**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

****

**BÁO CÁO BÀI TẬP LỚN MÔN DEEP LEARNING– CT282 LẦN THỨ 2**

***Đề tài***

**ỨNG DỤNG MẠNG HỌC SÂU LSTM**

**XÂY DỰNG MÔ HÌNH DỰ ĐOÁN GIÁ VÀNG**

***Giảng viên hướng dẫn:***

*PGS. TS. Phạm Nguyên Khang*

***Sinh viên thực hiện:***

*Nguyễn Minh Tuấn – B2017091*

*Nguyễn Hoàng Duy – B2016951*

*Cần Thơ, tháng 4 năm 2024*

**MỤC LỤC**

[1. Đặt vấn đề 4](#_Toc164111198)

[1.1. Lợi ích của việc dự đoán giá vàng 4](#_Toc164111199)

[1.2. Mục tiêu 4](#_Toc164111200)

[1.3. Hướng giải quyết 4](#_Toc164111201)

[1.3.1 Công cụ 4](#_Toc164111202)

[1.3.2. Nguồn dữ liệu 4](#_Toc164111203)

[1.3.3. Xử lý dữ liệu 5](#_Toc164111204)

[Chuẩn hóa dữ liệu: 5](#_Toc164111205)

[Tách tập dữ liệu: 6](#_Toc164111206)

[2. Xây dựng mạng học sâu LSTM 6](#_Toc164111207)

[3. Thử nghiệm các kiến trúc mạng 8](#_Toc164111208)

[Mô hình 1 8](#_Toc164111209)

[Kiến trúc mạng 8](#_Toc164111210)

[Tổng các tham số của mô hình: 8](#_Toc164111211)

[Thông số được sử dụng: 8](#_Toc164111212)

[Kết quả: 9](#_Toc164111213)

[Mô hình 2 11](#_Toc164111214)

[Kiến trúc mạng 11](#_Toc164111215)

[Tổng các tham số của mô hình 11](#_Toc164111216)

[Thông số được sử dụng: 11](#_Toc164111217)

[Kết quả: 12](#_Toc164111218)

[Mô hình 3 14](#_Toc164111219)

[Kiến trúc mạng 14](#_Toc164111220)

[Tổng các tham số của mô hình 14](#_Toc164111221)

[Thông số được sử dụng: 14](#_Toc164111222)

[Kết quả: 15](#_Toc164111223)

[4. Kết luận chung: 17](#_Toc164111224)

MỤC LỤC HÌNH ẢNH

[Hình 1.1: Logo Python 4](#_Toc164111329)

[Hình 2: Mô tả tập dữ liệu 5](#_Toc164111330)

[Hình 3: Tập dữ liệu trước khi chuẩn hóa 5](#_Toc164111331)

[Hình 4: Tập dữ liệu sau khi chuẩn hóa 6](#_Toc164111332)

[Hình 2.1: Sơ đồ mạng học sâu LSTM 7](#_Toc164111333)

[Hình 6: Tổng quan các tầng mô hình 1 8](#_Toc164111334)

[Hình 7: Đồ thị Training and Validation Loss của mô hình 1 9](#_Toc164111335)

[Hình 8: Biểu đồ dự đoán giá dựa trên mô hình 1 9](#_Toc164111336)

[Hình 9: Tổng quan các tầng mô hình 2 11](#_Toc164111337)

[Hình 10: Đồ thị Training and Validation Loss của mô hình 2 12](#_Toc164111338)

[Hình 11: Biểu đồ dự đoán giá dựa trên mô hình 2 12](#_Toc164111339)

[Hình 12: Tổng quan các tầng mô hình 3 14](#_Toc164111340)

[Hình 13: Đồ thị Training and Validation Loss của mô hình 3 15](#_Toc164111341)

[Hình 14: Biểu đồ dự đoán giá dựa trên mô hình 3 16](#_Toc164111342)

# Đặt vấn đề

Trong thị trường tài chính, việc dự đoán giá vàng là một trong những vấn đề khó khăn và quan trọng đối với các nhà đầu tư, doanh nghiệp và các tổ chức tài chính. Sự biến động không dự đoán được của giá vàng có thể gây ra những rủi ro và cơ hội đối với các bên liên quan.

## 1.1. Lợi ích của việc dự đoán giá vàng

Việc dự đoán giá vàng không chỉ là một công cụ hữu ích cho các nhà đầu tư và doanh nghiệp mà còn là một yếu tố quan trọng trong việc tạo ra giá trị trong thị trường tài chính. Bằng cách áp dụng các mô hình dự đoán tiên tiến như LSTM, chúng ta có thể tận dụng các thông tin lịch sử và xu hướng thị trường để đưa ra các dự đoán chính xác về giá vàng, từ đó giúp cải thiện quyết định đầu tư và quản lý rủi ro tài chính.

## 1.2. Mục tiêu

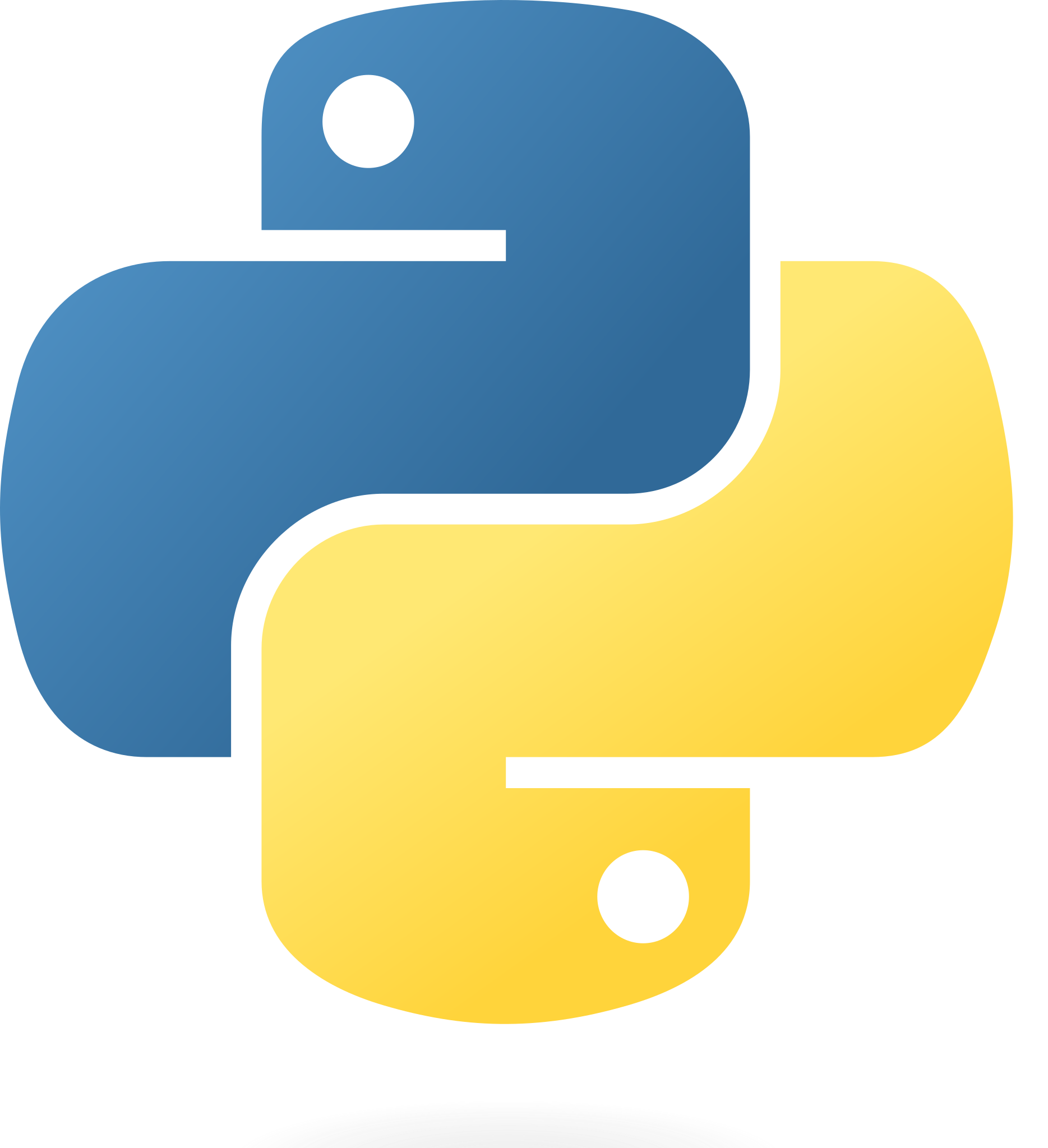
Mục tiêu của nghiên cứu này là áp dụng mạng học sâu LSTM (Long Short-Term Memory) để xây dựng một mô hình dự đoán giá vàng có khả năng đưa ra các dự báo chính xác về giá vàng trong tương lai.

## 1.3. Hướng giải quyết

Xây dựng model máy học ứng dụng mạng học sâu LSTM (Long Short-Term Memory) để giải quyết bài toán dự đoán giá vàng trong tương lai

### 1.3.1 Công cụ

**Python** là một ngôn ngữ lập trình bậc cao hỗ trợ mạnh cho mảng lập trình AI, học sâu, …

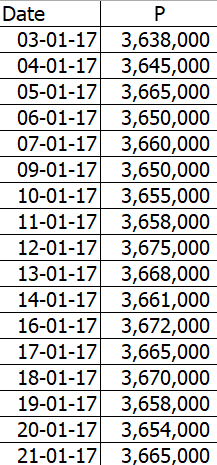


Hình 1.1: Logo Python

### 1.3.2. Nguồn dữ liệu

Dữ liệu được thu thập từ nhiều nguồn Kaggle, Google, Bing, … .

Dữ liệu gồm: 1492 dòng hiển thị giá vàng Việt Nam hằng ngày trong giai đoạn từ 01/01/2017 đến 31/12/2021 được công bố bởi Ngân hàng Thương mại Cổ phần Sài Gòn (SCB).

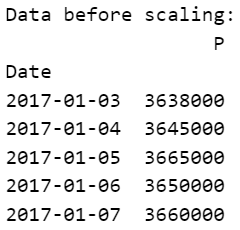


Hình 2: Mô tả tập dữ liệu

### 1.3.3. Xử lý dữ liệu

#### Chuẩn hóa dữ liệu:

Sử dụng **MinMaxScaler** là một công cụ trong sklearn để chuẩn hóa dữ liệu. Chuẩn hóa dữ liệu là quá trình biến đổi các giá trị của biến thành một phạm vi nhất định từ 0 đến 1. Việc đưa dữ liệu về cùng một khoảng này có thể giúp cho việc so sánh và phân tích dữ liệu dễ dàng hơn.

****

Hình 3: Tập dữ liệu trước khi chuẩn hóa

**A black numbers on a white background

Description automatically generated**

Hình 4: Tập dữ liệu sau khi chuẩn hóa

#### Tách tập dữ liệu:

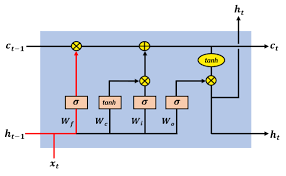
Để phân chia tập dữ liệu thành tập huấn luyện và tập validation để chuẩn bị cho việc xây dựng mô hình dự đoán. Độ dài của tập huấn luyện *(X\_train, y\_train)* được tính dựa trên số lượng bước thời gian liên tiếp *(timesteps)* và train\_len (tổng số dòng dữ liệu trừ đi độ dài tập kiểm tra, ở bài toán này là 180 dòng tương đương 3 tháng gần nhất để kiểm tra) Sau đó, một phần của tập huấn luyện được dùng làm tập validation *(X\_val, y\_val)* với tỷ lệ *val\_ratio*, để đánh giá hiệu suất của mô hình trong quá trình huấn luyện. Quá trình này giúp đảm bảo rằng mô hình được huấn luyện và đánh giá trên các tập dữ liệu độc lập, làm tăng tính đáng tin cậy của kết quả dự đoán.

# Xây dựng mạng học sâu LSTM

Trong lĩnh vực học sâu (deep learning), mạng học sâu LSTM (Long Short-Term Memory) là một dạng mạng nơ-ron có khả năng xử lý và học được các chuỗi dữ liệu thời gian một cách hiệu quả. LSTM được xây dựng dựa trên ý tưởng của mạng học sâu RNN (Recurrent Neural Network) nhưng được bổ sung thêm các cơ chế điều khiển để duy trì và quản lý thông tin trong quá trình lan truyền ngược gradient. Cấu trúc cơ bản của một LSTM bao gồm các thành phần chính sau:

* 1. ***Cổng quên:*** *Quyết định thông tin nào sẽ được giữ lại và thông tin nào sẽ bị loại bỏ từ bộ nhớ trước đó.*
  2. ***Cổng đầu vào:*** *Xác định thông tin cần thiết để lưu lại*
  3. ***Tầng tanh:*** *Tầng này có chức năng tạo ra một phạm vi giá trị cho các cập nhật tiềm năng cho bộ nhớ.*
  4. ***Cổng đầu ra:*** *Quyết định thông tin nào sẽ được gửi đến phần tử kế tiếp của chuỗi hoặc đầu ra của mạng.*

Các cổng này cho phép LSTM học và lưu trữ thông tin từ các bước thời gian trước đó của chuỗi dữ liệu một cách linh hoạt và hiệu quả.



Hình 2.1: Sơ đồ mạng học sâu LSTM

**Bước 1:** Xác định thông tin không cần thiết bằng cách sử dụng cổng quên.

ft = σ(Wf [ht-1, xt] + bf )

**Bước 2:**

Xác định thông tin cần thiết để lưu lại

Dùng 2 tầng Cổng đầu vào và Tầng Tanh

it​=σ(Wi[ht-1, xt] + bi)

c~t = tanh(Wc[ht-1, xt] + bc)

**Bước 3:**

Cập nhật trạng thái mới từ trạng thái cũ (ct-1) và ứng viên mới (c~t)

ct = ft \* ct-1 + it\*c~t

**Bước 4:**

Xác định đầu ra ht

Đưa đầu ra trước đó ht-1 và đầu vào hiện tại xt qua cổng sigmod

Nhân kết quả với tanh (trạng thái hiện tại - ct)

ot = σ(Wo[ht-1, xt] + bo)

ht = ot\*tanh(ct)

# Thử nghiệm các kiến trúc mạng

## Mô hình 1

### Kiến trúc mạng

* Tầng LSTM đầu tiên với 128 đơn vị LSTM, nhận đầu vào là một chuỗi dữ liệu có độ dài 60 (timesteps) và trả về đầu ra có cùng kích thước với số lượng đơn vị LSTM là 128.
* Tầng LSTM thứ hai với 64 đơn vị LSTM, nhận đầu vào từ lớp LSTM trước đó và trả về đầu ra có kích thước 64.
* Tầng fully connected (Dense) với 30 đơn vị, nhận đầu vào từ lớp LSTM trước đó và thực hiện các phép tính tuyến tính.
* Tầng fully connected (Dense) cuối cùng với đầu ra là 1 đơn vị, thực hiện dự đoán cuối cùng.

A screenshot of a computer

Description automatically generated

Hình 2: Tổng quan các tầng mô hình 1

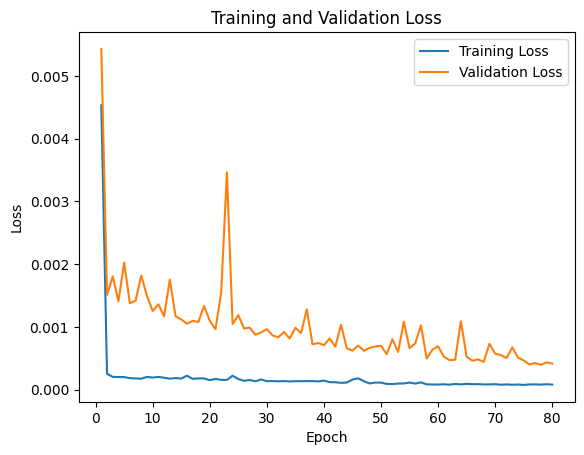
### Tổng các tham số của mô hình:

* Total params: 117.949 (460.74 KB)
* Trainable params: 117.949 (460.74 KB)
* Non-trainable params: 0 (0.00 Byte)

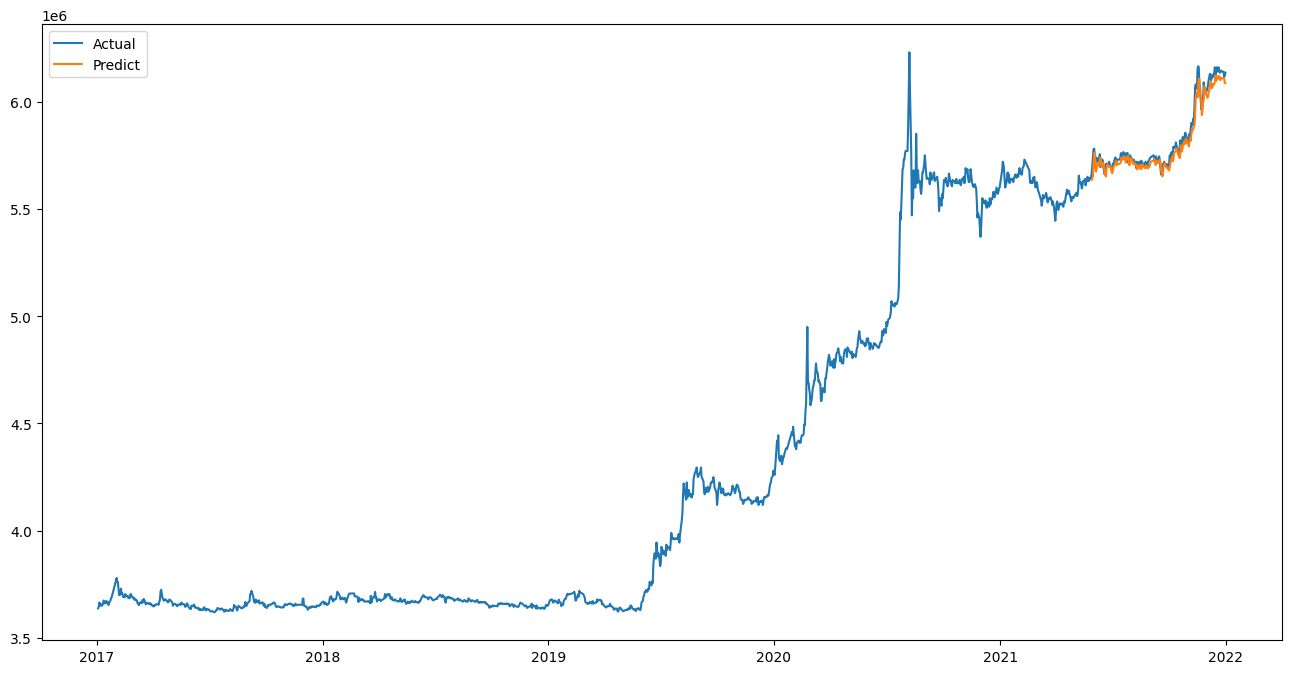
### Thông số được sử dụng:

* Hàm lỗi: Mean Squared Error
* Optimizer: Adam
* Epochs: 80
* Batch size: 32
* Learning rate: 0.001

### Kết quả:



Hình 3: Đồ thị Training and Validation Loss của mô hình 1



Hình 4: Biểu đồ dự đoán giá dựa trên mô hình 1

**Kết quả đánh giá trên tập test:**

* Final Training Loss: 7.920836651464924e-05
* Kiểm tra chỉ số **RMSE** trên tập test: 26939.34

**Đánh giá:**

* + Biểu đồ loss: sự hội tụ giữa train và validation biểu đồ cho thấy mô hình không gặp phải vấn đề lớn về overfitting hoặc underfitting. Điều này làm cho mô hình có khả năng tổng quát hóa tốt trên dữ liệu mới..
  + Kết quả kiểm thử model 1 cho thấy rằng nó hiệu quả trong việc dự đoán giá vàng, đặc biệt là trong việc xử lý các chu kỳ tăng giảm của giá cả. Điều này cho thấy mô hình đã học được các quy luật và mẫu trong dữ liệu một cách hiệu quả

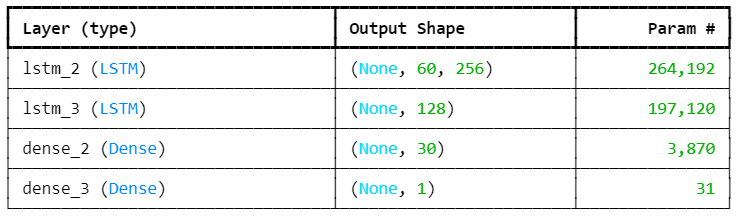
**Kết luận:**

Mặc dù biểu đồ loss ổn định, nhưng sự biến thiên đột ngột trong dữ liệu có thể làm cho mô hình không thể dự đoán được theo sát. Có thể cần xem xét các phương pháp khác nhau để xử lý dữ liệu đặc biệt này, chẳng hạn như việc áp dụng các kỹ thuật xử lý tín hiệu hoặc dữ liệu thêm thông tin bổ sung để giúp mô hình hiểu được mô hình hơn.

## Mô hình 2

### Kiến trúc mạng

* Tầng LSTM đầu tiên với 256 đơn vị LSTM, nhận đầu vào là một chuỗi dữ liệu có độ dài 60 (timesteps) và trả về đầu ra có cùng kích thước với số lượng đơn vị LSTM là 256.
* Tầng LSTM thứ hai với 128 đơn vị LSTM, nhận đầu vào từ lớp LSTM trước đó và trả về đầu ra có kích thước 128.
* Tầng fully connected (Dense) với 30 đơn vị, nhận đầu vào từ lớp LSTM trước đó và thực hiện các phép tính tuyến tính.
* Tầng fully connected (Dense) cuối cùng với đầu ra là 1 đơn vị, thực hiện dự đoán cuối cùng.



Hình 5: Tổng quan các tầng mô hình 2

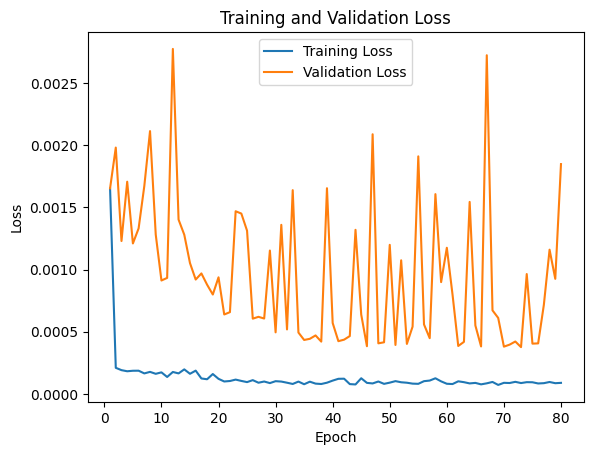
### Tổng các tham số của mô hình

* Total params: 465.213 (1.76 MB)
* Trainable params: 465.213 (1.76 MB)
* Non-trainable params: 0 (0.00 Byte)

### Thông số được sử dụng:

* Hàm lỗi: Mean Squared Error
* Optimizer: Adam
* Epochs: 80
* Batch size: 16
* Learning rate: 0.001

### Kết quả:



Hình 6: Đồ thị Training and Validation Loss của mô hình 2

A blue line graph on a white background

Description automatically generated

Hình 7: Biểu đồ dự đoán giá dựa trên mô hình 2

**Kết quả đánh giá trên tập test:**

* Final Training Loss: 8.915729995351285e-05
* Kiểm tra chỉ số **RMSE** trên tập test: 123770.50

**Đánh giá:**

* + - Biểu đồ loss: kết quả train loss rất thấp nhưng train validation biến thiên liên tục, chứng tỏ model 2 có vấn đề. Không thể hội tụ được, chứng tỏ mô hình không đạt được kết quả tốt có thể overfiting
    - Kết quả kiểm thử model 2: cho ta thấy model 2 không thể dự đoán tốt được so với model 1, đường kết quả dự đoán lệch khá nhiều so với đường thực tế.

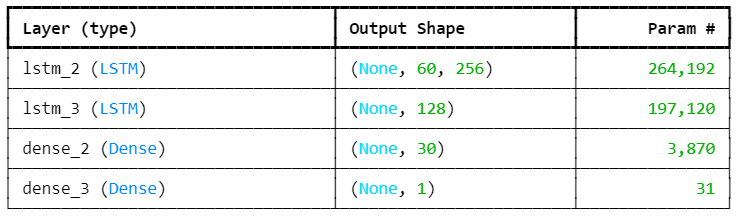
**Kết luận:**

Do model 2 đã gia tăng bộ tham số lên so với model 1, khiến cho mô hình có thể quá phức tạp so để xử lý bộ dữ liệu. Điều này cho thấy mô hình đang học cách fit dữ liệu huấn luyện tốt hơn nhưng không tổng quát hóa được với dữ liệu mới (tập validation). Điều này có thể xuất phát từ mô hình quá phức tạp, dẫn đến việc mô hình "nhớ" dữ liệu huấn luyện thay vì học được các mẫu tổng quát.

## Mô hình 3

### Kiến trúc mạng

* Tầng LSTM đầu tiên với 256 đơn vị LSTM, nhận đầu vào là một chuỗi dữ liệu có độ dài 60 (timesteps) và trả về đầu ra có cùng kích thước với số lượng đơn vị LSTM là 256.
* Tầng LSTM thứ hai với 128 đơn vị LSTM, nhận đầu vào từ lớp LSTM trước đó và trả về đầu ra có kích thước 128.
* Tầng fully connected (Dense) với 30 đơn vị, nhận đầu vào từ lớp LSTM trước đó và thực hiện các phép tính tuyến tính.
* Tầng fully connected (Dense) cuối cùng với đầu ra là 1 đơn vị, thực hiện dự đoán cuối cùng.
* Kiến trúc của mô hình 3 lấy lại kiến trúc của mô hình 2 nhưng áp dụng **Early Stopping**.
  + Early stopping là một kỹ thuật trong quá trình huấn luyện mạng nơ-ron để ngừng huấn luyện sớm (early stopping) dựa trên sự giám sát của hiệu suất trên tập validation. Mục đích chính của early stopping là ngăn chặn hiện tượng overfitting và tăng tính tổng quát của mô hình.



Hình 8: Tổng quan các tầng mô hình 3

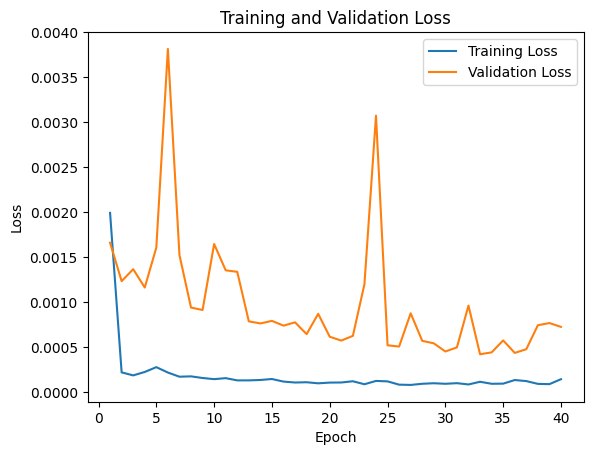
### Tổng các tham số của mô hình

* Total params: 465.213 (1.76 MB)
* Trainable params: 465.213 (1.76 MB)
* Non-trainable params: 0 (0.00 Byte)

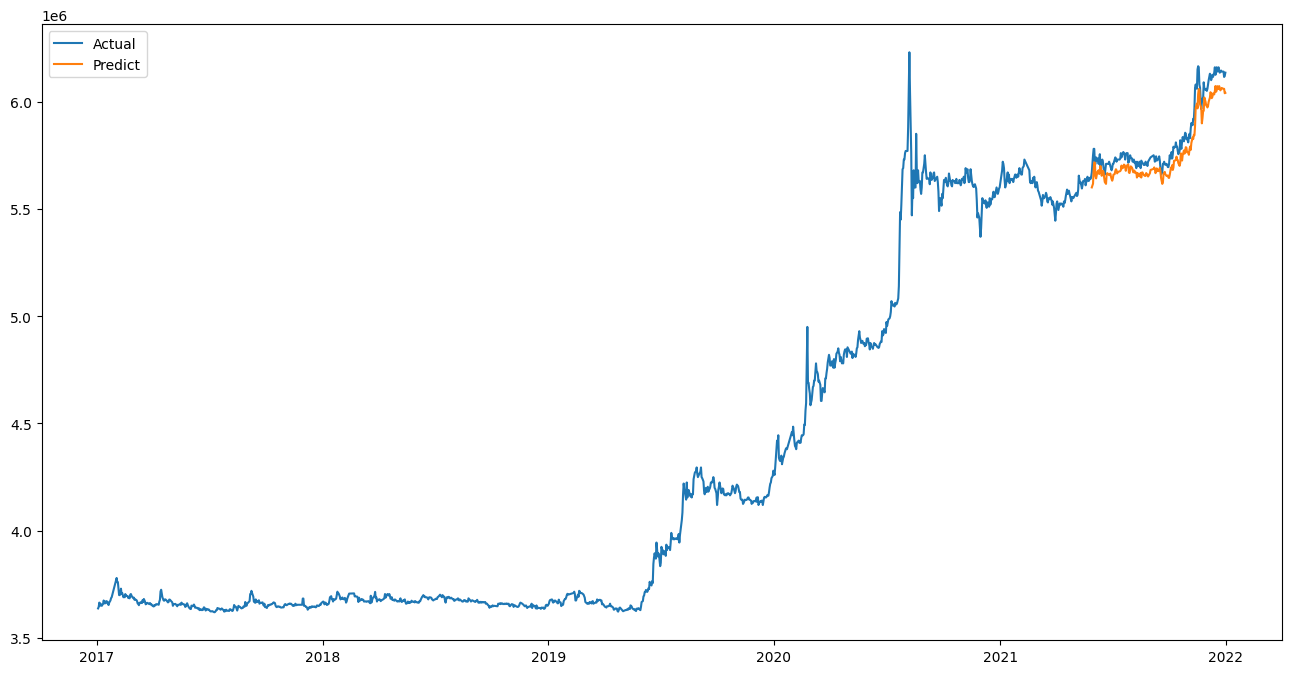
### Thông số được sử dụng:

* Hàm lỗi: Mean Squared Error
* Optimizer: Adam
* Epochs: 80
* Batch size: 16
* Learning rate: 0.001
* **Call backs: Early stopping**

### Kết quả:



Hình 9: Đồ thị Training and Validation Loss của mô hình 3



Hình 10: Biểu đồ dự đoán giá dựa trên mô hình 3

**Kết quả đánh giá trên tập test:**

* Final Training Loss: 0.00014397641643881798
* Kiểm tra chỉ số **RMSE** trên tập test: 69672.89

**Đánh giá:**

* + - Biểu đồ loss: Model 3 đã có sự cải thiện đáng kể so với Model 2 nhờ sử dụng kỹ thuật early stopping. Biểu đồ loss cho thấy sự giảm thiểu loss trên cả tập huấn luyện và tập validation. Mặc dù loss trên tập validation vẫn cao và biến thiên nhiều hơn so với tập huấn luyện, early stopping đã giúp ngăn chặn quá trình overfitting bằng cách dừng lại ở epoch thứ 41 để lưu lại một mô hình với kết quả tốt nhất trước khi sự overfitting xảy ra.
    - Kết quả kiểm thử Model 3: cho thấy sự cải thiện đáng kể so với Model 2, với đường dự đoán đã bám sát hơn với đường giá thực tế. Điều này cho thấy mô hình đã học được các mẫu và quy luật trong dữ liệu một cách hiệu quả.

**Kết luận:**

Mặc dù đã thêm early stopping để ngăn chặn overfitting tuy nhiên kết quả ở model 3 vẫn chưa thực sự tối ưu, chứng tỏ bộ tham số ở model 3 vẫn chưa phù hợp với bộ dữ liệu này. Cần phải kết hợp giảm bớt bộ tham số, giảm bớt độ phức tạp của kiến trúc để model hoạt động tốt hơn

# Kết luận chung:

- Chỉ số RMSE ở Model 1 tốt nhất:

* Model 1: 26939.34
* Model 2: 123770.50
* Model 3: 69672.89

- Biểu đồ Loss

* Biểu đồ Loss ở model 1 tuy có sự biến thiên nhưng nhìn chung vẫn không ảnh hưởng nhiều, vẫn có hội tụ.
* Biểu đồ Loss ở model 2 với bộ tham số không phù hợp, nhiều neurons ở các tầng dẫn đến overfitting.
* Biểu đồ Loss ở model 3: nhìn chung có sự hội tụ hơn so với model 2, nhưng vẫn còn bị biến thiên ở validation loss khá nhiều.

- Kết quả thực nghiệm:

* Model 1: kết quả khá tốt khi đường dự đoán bám gần sát với thực tế.
* Model 2: kết quả không tốt như model 1, kết quả bị lệch nhiều so với thực tế.
* Model 3: cải thiện so với model 2 nhờ áp dụng early stopping. Tuy nhiên do bộ tham số không phù hợp, theo suy đoán bộ tham số sử dụng ở model 2 và 3 đang quá lớn so với dữ liệu và bài toán cần xử lý khiến cho model bị overfitting