



**ĐẠI HỌC BÁCH KHOA HÀ NỘI**

HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY



**KHOA TOÁN - TIN**

Faculty Of Applied Mathematics And Informatics

# **Đề tài: Dự báo giá chứng khoán dựa trên mô hình RNN và LSTM**

Giảng viên hướng dẫn: PGS.TS. Nguyễn Đình Hân

Sinh viên thực hiện: Lê Ngọc Hà

MSSV: 20216922

Lớp: MI2 01 - K66

Hà Nội, 01/2025

# Mục lục

- 1 Bài toán dự báo giá chứng khoán
- 2 Cơ sở lý thuyết
- 3 Xây dựng mô hình
- 4 Kết quả và đánh giá

# Mục lục

**1** Bài toán dự báo giá chứng khoán

2 Cơ sở lý thuyết

3 Xây dựng mô hình

4 Kết quả và đánh giá

# Bài toán dự báo giá chứng khoán

Trong bối cảnh phát triển mạnh mẽ của thị trường tài chính, dự báo giá chứng khoán trở thành nhu cầu cấp thiết, giúp hỗ trợ đưa ra các quyết định đầu tư nhằm tối đa hoá lợi nhuận.

Một số phương pháp dự báo giá chứng khoán:

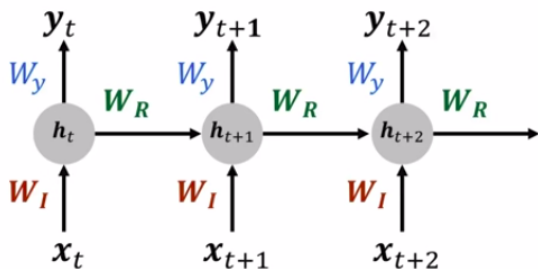
- Phân tích cơ bản.
- Phân tích kỹ thuật.
- Mô hình chuỗi thời gian.
- Phương pháp học máy.



# Mục lục

- 1 Bài toán dự báo giá chứng khoán
- 2 Cơ sở lý thuyết**
- 3 Xây dựng mô hình
- 4 Kết quả và đánh giá

# Mô hình Recurrent Neural Network



Công thức của mô hình RNN được biểu diễn như sau:

$$h_t = f(W_r h_{t-1} + W_i x_t + b) \quad (1)$$

$$y_t = W_y h_t \quad (2)$$

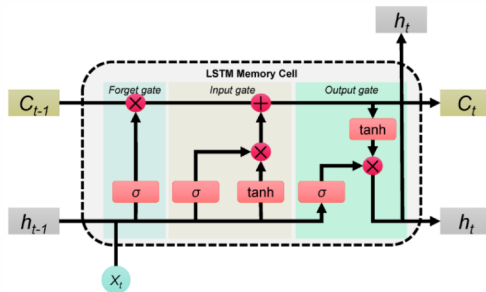
**Input:** Các giá trị đầu vào  $x_t$

**Output:** Các giá trị dự đoán  $y_t$

Trong đó:

- $x_t$ : Giá trị đầu vào
- $h_t$ : Trạng thái ẩn
- $y_t$ : Giá trị đầu ra
- $f$ : Hàm kích hoạt
- $W_r$ : Trọng số giữa  $h_{t-1}$  và  $h_t$
- $W_i$ : Trọng số giữa  $h_t$  và  $x_t$
- $W_y$ : Trọng số giữa  $h_t$  và  $y_t$
- $b$ : Hệ số bias riêng

# Mô hình Long Short - Term Memory



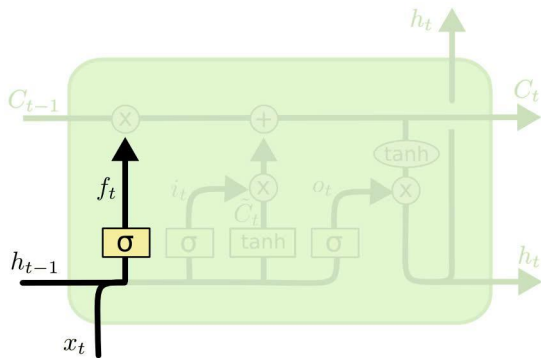
Cấu tạo của mô hình LSTM bao gồm các thành phần sau:

- 3 cổng chính: Forget Gate, Input Gate, Output Gate.
- Cell State (C): Thanh trạng thái bộ nhớ.
- Hidden State (h): Trạng thái ẩn.

# Mô hình Long Short - Term Memory

**Bước 1:** Input là các giá trị  $x_t, h_{t-1}$ .

**Bước 2:** Giá trị  $x_t, h_{t-1}$  được xử lý tại Forget Gate.



$$f_t = \sigma(U_f * x_t + W_f * h_{t-1} + b_f) \quad (3)$$

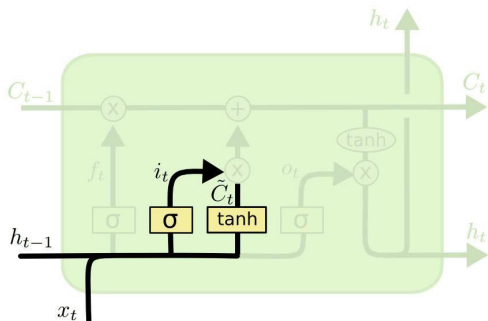
Trong đó:

- $f_t$ : Đầu ra của Forget Gate tại  $t$
- $\sigma$ : Hàm kích hoạt sigmoid
- $U_f$ : Trọng số của đầu vào  $x_t$
- $x_t$ : Đầu vào ở bước hiện tại
- $W_f$ : Trọng số của Forget Gate
- $h_{t-1}$ : Trạng thái ẩn tại  $t - 1$



# Mô hình Long Short - Term Memory

**Bước 3:** Tại Input Gate sẽ quyết định thông tin nào cần được thêm vào trạng thái bộ nhớ  $c$ .



Trong đó:

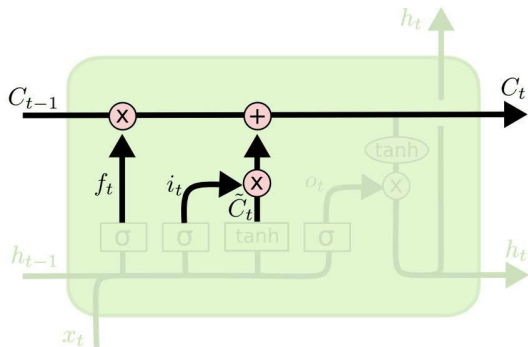
- $i_t$ : Đầu ra của Input Gate tại  $t$
- $U_i$ : Trọng số giữa  $x_t$  và Input Gate
- $W_i$ : Trọng số giữa  $h_{t-1}$  và Input Gate
- $\tilde{c}_t$ : Giá trị điều chỉnh
- $\tanh$ : Hàm kích hoạt
- $U_c$ : Trọng số giữa  $x_t$  và  $\tilde{c}_t$
- $W_c$ : Trọng số giữa  $h_{t-1}$  và  $\tilde{c}_t$

$$i_t = \sigma(U_i * x_t + W_i * h_{t-1} + b_i) \quad (4)$$

$$\tilde{c}_t = \tanh(U_c * x_t + W_c * h_{t-1} + b_c) \quad (5)$$

## Mô hình Long Short - Term Memory

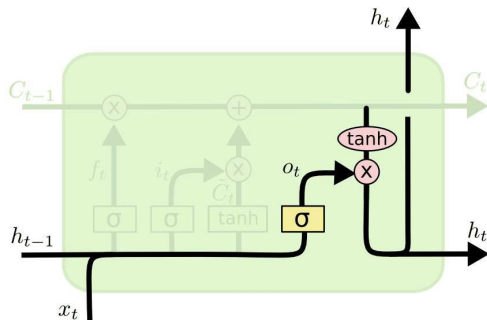
**Bước 3:** Tại Input Gate sẽ quyết định thông tin nào cần được thêm vào trạng thái bộ nhớ  $c$ .



$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t \quad (6)$$

## Mô hình Long Short - Term Memory

**Bước 4:** Tại Output Gate điều chỉnh lượng thông tin được đề xuất ra ngoài  $y_t$  và truyền đến trạng thái ẩn tiếp theo  $h_t$ .



$$o_t = \sigma(U_o x_t + W_o h_{t-1} + b_o) \quad (7)$$

$$h_t = o_t \tanh(c_t) \quad (8)$$

Trong đó:

- $o_t$ : Đầu ra của Output Gate tại  $t$
- $U_o$ : Trọng số giữa  $x_t$  và Output Gate
- $W_o$ : Trọng số giữa  $h_t$  và Output Gate

# Mục lục

- 1 Bài toán dự báo giá chứng khoán
- 2 Cơ sở lý thuyết
- 3 Xây dựng mô hình**
- 4 Kết quả và đánh giá

## Phát biểu bài toán

Trong bài toán này sẽ tập trung dự báo giá chứng khoán với mã cổ phiếu FPT của Công ty Cổ phần Viễn thông FPT đã được giao dịch trên sàn HOSE. Đồng thời, sử dụng thư viện *vnstock* để thu thập dữ liệu.



# Mô tả và tiền xử lý dữ liệu

Bộ dữ liệu là thông tin về giá và khối lượng giao dịch hàng ngày của Công ty cổ phần Viễn thông FPT trong khoảng thời gian 5 năm từ 20/11/2019 đến 20/11/2024.

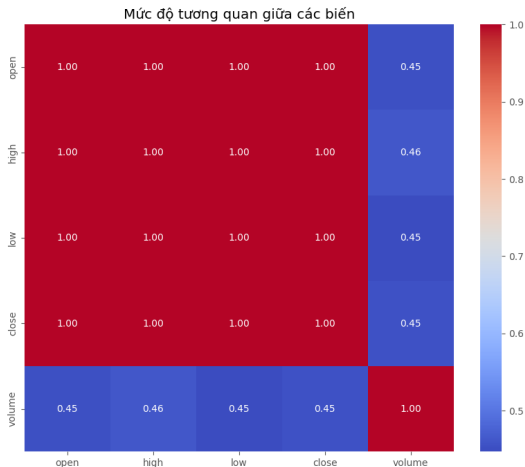
	time	open	high	low	close	volume
0	2019-11-20	24.64	24.64	23.96	24.05	2291760
1	2019-11-21	24.05	24.18	23.67	23.71	3303950
2	2019-11-22	23.84	24.18	23.50	23.71	1625820
3	2019-11-25	23.75	24.13	23.62	23.96	1087990
4	2019-11-26	24.13	24.22	24.05	24.18	968690
...	...	...	...	...	...	...
1247	2024-11-15	134.37	134.76	131.69	132.98	6574421
1248	2024-11-18	133.08	133.67	131.59	133.08	3910975
1249	2024-11-19	132.58	132.68	129.00	129.10	7652727
1250	2024-11-20	129.10	131.49	125.83	131.49	8474055
1251	2024-11-21	131.49	132.08	130.10	132.08	3183165

[1252 rows x 6 columns]

Tiền xử lý dữ liệu bao gồm:

- Chia tập dữ liệu: Test, Validation, Train.
- Chuẩn hoá dữ liệu về phạm vi [0,1].

# Mức độ tương quan giữa các biến



Mức độ tương quan của các biến còn lại so với biến "close":

high	0.999762
low	0.999697
open	0.999454
volume	0.454607

# Xây dựng và huấn luyện mô hình

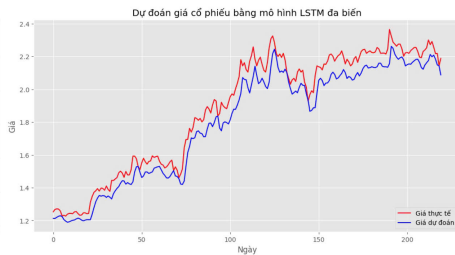
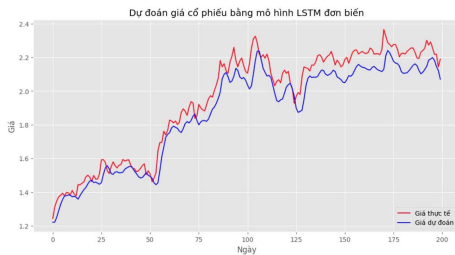
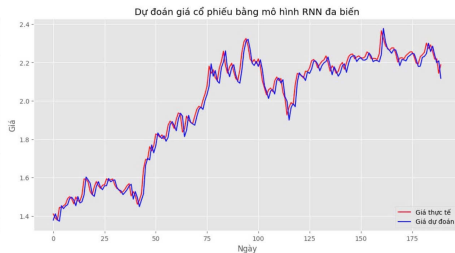
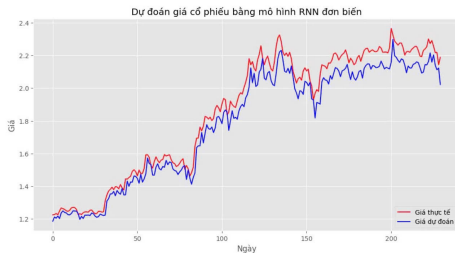
Các siêu tham số	RNN đơn biến	RNN đa biến	LSTM đơn biến	LSTM đa biến
timesteps	20	60	50	30
hidden layers	[50,40]	[35,35]	[20,25]	[35,25]
learning rate	0.003	0.006	0.002	0.003
batch size	64	64	64	64
epochs	150	300	300	250



# Mục lục

- 1 Bài toán dự báo giá chứng khoán
- 2 Cơ sở lý thuyết
- 3 Xây dựng mô hình
- 4 Kết quả và đánh giá**

# Kết quả mô hình



## Kết quả mô hình

Mô hình	Biến sử dụng	MSE	RMSE	MAPE	$R^2$
RNN đơn biến	close	0.0068	0.0823	0.0364	0.9518
RNN đa biến	high, volume, close	0.0014	0.0380	0.0151	0.9827
LSTM đơn biến	close	0.0082	0.0907	0.0388	0.9157
LSTM đa biến	high, volume, close	0.0086	0.0929	0.0433	0.9332

**Bảng:** Chỉ số đánh giá kết quả dự báo giá chứng khoán cổ phiếu FPT

## Đánh giá mô hình

- Các mô hình đa biến mang lại kết quả cao hơn so với mô hình đơn biến vì nó có thể khai thác được nhiều thông tin hơn từ dữ liệu, giúp mô hình nhận diện tốt hơn các tín hiệu phức tạp ảnh hưởng đến giá cổ phiếu.
- Trong bộ dữ liệu này không yêu cầu khả năng ghi nhớ dài hạn của các chuỗi nên trong trường hợp này, RNN có thể phát huy tối đa ưu điểm vì nó có khả năng xử lý các chuỗi dữ liệu có độ dài vừa phải mà không cần đến khả năng ghi nhớ lâu dài của LSTM.
- Tuy nhiên, nếu sau này chuỗi dữ liệu dài hơn hoặc phức tạp hơn thì LSTM có thể sẽ phát huy được hiệu quả cao hơn.

## Kết quả đạt được

- Trang bị thêm các kiến thức về bài toán dự báo giá chứng khoán.
- Tìm hiểu về những kiến thức toán học như chuỗi thời gian, mạng nơ-ron, các siêu tham số, ôn lại các chỉ số đánh giá mô hình. Nghiên cứu chi tiết về hai mô hình là RNN và LSTM cùng các ưu, nhược điểm của từng mô hình.
- Tiến hành xây dựng và huấn luyện mô hình, xử lý các vấn đề vanishing gradient, overfitting trong quá trình huấn luyện mô hình. Kết hợp với các thư viện để tìm các siêu tham số giúp mô hình tối ưu hơn.
- So sánh và đánh giá kết quả mà hai mô hình RNN và LSTM đem lại.

## Hướng phát triển tương lai

- Tối ưu các phương pháp khác nhau để thời gian tìm các siêu tham số nhanh hơn.
- Thử nghiệm mô hình trên nhiều bộ dữ liệu chứng khoán của các công ty khác nhau.
- Cải tiến mô hình hơn như làm giàu dữ liệu, đồng thời xây dựng một cơ sở dữ liệu để lưu trữ và phân tích giá chứng khoán...

## Tài liệu tham khảo

- [1] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, Time series analysis: Forecasting and control, 5th. Wiley, 2015.
- [2] I. Goodfellow, Y. Bengio, and A. Courville, Deep learning. MIT Press, 2016.
- [3] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," Nature, vol. 323, no. 6088, pp. 533–536, 1986. DOI: 10.1038/323533a0.
- [4] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," Neural Computation, vol. 1, no. 2, pp. 270–280, 1989. DOI: 10.1162/neco.1989.1.2.270.
- [5] T. Mikolov, M. Karafiát, L. Burget, and J. Cernocký, "Recurrent neural network based language model," Interspeech, pp. 1045–1048, 2010. DOI: 10.21437/Interspeech.2010-285.
- [6] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997. DOI: 10.1162/neco.1997.9.8.1735.

## Tài liệu tham khảo

- [7] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," IEEE Transactions on Neural Networks, vol. 5, no. 2, pp. 157–166, 1994. DOI: 10.1109/72.279181.
- [8] F. A. Gers and J. Schmidhuber, "Learning to forget: Continual prediction with lstm," in Proceedings of the International Conference on Artificial Neural Networks (ICANN), Springer, 2000, pp. 850–855.
- [9] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," International Conference on Learning Representations (ICLR), 2015. [Online]. Available: <https://arxiv.org/abs/1412.6974>
- [10] J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in Advances in Neural Information Processing Systems, vol. 24, 2011, pp. 2546–2554. [Online]. Available: <https://papers.nips.cc/paper/4443-algorithms-for-hyperparameter-optimization.pdf>.



**CẢM ƠN THẦY/CÔ VÀ CÁC BẠN ĐÃ LẮNG NGHE!**