Đại học quốc gia Thành phố Hồ Chí Minh

Trường Đại học Công nghệ thông tin

Khoa công nghệ phần mềm

Báo cáo cuối kì môn Một số thuật toán thông minh

Dự đoán giá vàng sử dụng Tensorflow

Giảng viên hướng dẫn ThS.Nguyễn Công Hoan

Sinh viên thực hiện Phạm Hùng Vỹ - 15521037
Trần Phú Vinh- 15521020
Đào Đức Huy - 15520295
Phan Hữu Chí - 15520065

20/06/2019
Version: 1.0

Lời cảm ơn

Nhóm chúng em hoàn thành được tốt đồ án môn học này, không thể không nói đến công lao của thầy Nguyễn Công Hoan. Chúng em xin chân thành cảm ơn thầy đã tận tâm hướng dẫn chúng em. Bên cạnh đó, nhóm cũng xin gửi lời cảm ơn chân thành đến các anh chị khóa trên, các bạn trong và ngoài lớp đã sẵn lòng chia sẻ tài liệu cũng như kinh nghiệm từng trải của bản thân để nhóm chúng em học tập và tránh mắc những sai lầm, tiết kiệm được thời gian trong quá trình thực hiện đồ án.

Tuy nhiên, do kiến thức và khả năng của chúng em còn nhiều hạn chế, do đó không tránh khỏi những thiếu sót, yếu kém. Chúng em rất mong nhận được những ý kiến đóng góp quý báu của thầy cô và các bạn học cùng lớp để đồ án được hoàn thiện hơn và rút ra được kinh nghiệm. Sau cùng, chúng em xin kính chúc quý thầy cô ở Khoa Công nghệ Phần mềm, đặc biệt là thầy Nguyễn Công Hoan thật dồi dào sức khỏe để tiếp tục thực hiện sứ mệnh cao đẹp của mình là truyền đạt kiến thức cho thế hệ mai sau. Nhóm xin chân thành cảm ơn!

Mục lục

1 Một số khái niệm

1.1 Dự báo

1.1.1 Khái niệm

Dự báo đã hình thành từ đầu những năm 60 của thế kỉ 20. Khoa học dự báo với tư cách một ngành khoa học độc lập có hệ thống lí luận, phương pháp luận và phương pháp hệ riêng nhằm nâng cao tính hiệu quả của dự báo. Người ta thường nhấn mạnh rằng một phương pháp tiếp cận hiệu quả đối với dự báo là phần quan trọng trong hoạch định. Khi các nhà quản trị lên kế hoạch, trong hiện tại họ xác định hướng tương lai cho các hoạt động mà họ sẽ thực hiện. Bước đầu tiên trong hoạch định là dự báo hay là ước lượng nhu cầu tương lai cho sản phẩm hoặc dịch vụ và các nguồn lực cần thiết để sản xuất sản phẩm hoặc dịch vụ đó. Như vậy, dự báo là một khoa học và nghệ thuật tiên đoán những sự việc sẽ xảy ra trong tương lai, trên cơ sở phân tích khoa học về các dữ liệu đã thu thập được. Khi tiến hành dự báo ta căn cứ vào việc thu thập xử lý số liệu trong quá khứ và hiện tại để xác định xu hướng vận động của các hiện tượng trong tương lai nhờ vào một số mô hình toán học. Dự báo có thể là một dự đoán chủ quan hoặc trực giác về tương lai. Nhưng để cho dự báo được chính xác hơn, người ta cố loại trừ những tính chủ quan của người dự báo. Ngày nay, dự báo là một nhu cầu không thể thiếu được của mọi hoạt động kinh tế - xác hội, khoa học - kỹ thuật, được tất cả các ngành khoa học quan tâm nghiên cứu.

1.1.2 Ý nghĩa

- Dùng để dự báo các mức độ tương lai của hiện tượng, qua đó giúp các nhà quản trị doanh nghiệp chủ động trong việc đề ra các kế hoạch và các quyết định cần thiết phục vụ cho quá trình sản xuất kinh doanh, đầu tư, quảng bá, quy mô sản xuất, kênh phân phối sản phẩm, nguồn cung cấp tài chính... và chuẩn bị đầy đủ điều kiện cơ sở vật chất, kỹ thuật cho sự phát triển trong thời gian tới (kế hoạch cung cấp các yếu tố đầu vào như: lao động, nguyên vật liệu, tư liệu lao động... cũng như các yếu tố đầu ra dưới dạng sản phẩm vật chất và dịch vụ).

- Trong các doanh nghiệp nếu công tác dự báo được thực hiện một cách nghiêm túc còn tạo điều kiện nâng cao khả năng cạnh tranh trên thị trường.

- Dự báo chính xác sẽ giảm bớt mức độ rủi ro cho doanh nghiệp nói riêng và toàn bộ nền kinh tế nói chung.

- Dự báo chính xác là căn cứ để các nhà hoạch định các chính sách phát triển kinh tế văn hoá xã hội trong toàn bộ nền kinh tế quốc dân

- Nhờ có dự báo các chính sách kinh tế, các kế hoạch và chương trình phát triển kinh tế được xây dựng có cơ sở khoa học và mang lại hiệu quả kinh tế cao.

- Nhờ có dự báo thường xuyên và kịp thời, các nhà quản trị doanh nghiệp có khả năng kịp thời đưa ra những biện pháp điều chỉnh các hoạt động kinh tế của đơn vị mình nhằm thu được hiệu quả sản xuất kinh doanh cao nhất.

1.1.3 Vai trò

- Dự báo tạo ra lợi thế cạnh tranh

- Công tác dự báo là một bộ phận không thể thiếu trong hoạt động của các doanh nghiệp, trong từng phòng ban như: phòng Kinh doanh hoặc Marketing, phòng Sản xuất hoặc phòng Nhân sự, phòng Kế toán – tài chính.

1.1.4 Các loại dự báo

Căn cứ vào độ dài thời gian dự báo: Dự báo có thể phân thành ba loại

- Dự báo dài hạn: Là những dự báo có thời gian dự báo từ 5 năm trở lên. Thường dùng để dự báo những mục tiêu, chiến lược về kinh tế chính trị, khoa học kỹ thuật trong thời gian dài ở tầm vĩ mô.

- Dự báo trung hạn: Là những dự báo có thời gian dự báo từ 3 đến 5 năm. Thường phục vụ cho việc xây dựng những kế hoạch trung hạn về kinh tế văn hoá xã hội... ở tầm vi mô và vĩ mô.

- Dự báo ngắn hạn: Là những dự báo có thời gian dự báo dưới 3 năm, loại dự báo này thường dùng để dự báo hoặc lập các kế hoạch kinh tế, văn hoá, xã hội chủ yếu ở tầm vi mô và vĩ mô trong khoảng thời gian ngắn nhằm phục vụ cho công tác chỉ đạo kịp thời.

Cách phân loại này chỉ mang tính tương đối tuỳ thuộc vào từng loại hiện tượng để quy định khoảng cách thời gian cho phù hợp với loại hiện tượng đó: ví dụ trong dự báo kinh tế, dự báo dài hạn là những dự báo có tầm dự báo trên 5 năm, nhưng trong dự báo thời tiết, khí tượng học chỉ là một tuần. Thang thời gian đối với dự báo kinh tế dài hơn nhiều so với thang thời gian dự báo thời tiết. Vì vậy, thang thời gian có thể đo bằng những đơn vị thích hợp (ví dụ: quý, năm đối với dự báo kinh tế và ngày đối với dự báo dự báo thời tiết).

Dựa vào các phương pháp dự báo: Dự báo có thể chia thành 3 nhóm

- Dự báo bằng phương pháp chuyên gia: Loại dự báo này được tiến hành trên cơ sở tổng hợp, xử lý ý kiến của các chuyên gia thông thạo với hiện tượng được nghiên cứu, từ đó có phương pháp xử lý thích hợp đề ra các dự đoán, các dự đoán này được cân nhắc và đánh giá chủ quan từ các chuyên gia. Phương pháp này có ưu thế trong trường hợp dự đoán những hiện tượng hay quá trình bao quát rộng, phức tạp, chịu sự chi phối của khoa học - kỹ thuật, sự thay đổi của môi trường, thời tiết, chiến tranh trong khoảng thời gian dài. Một cải tiến của phương pháp Delphi – là phương pháp dự báo dựa trên cơ sở sử dụng một tập hợp những đánh giá của một nhóm chuyên gia. Mỗi chuyên gia được hỏi ý kiến và rồi dự báo của họ được trình bày dưới dạng thống kê tóm tắt. Việc trình bày những ý kiến này được thực hiện một cách gián tiếp (không có sự tiếp xúc trực tiếp) để tránh những sự tương tác trong nhóm nhỏ qua đó tạo nên những sai lệch nhất định trong kết quả dư báo. Sau đó người ta yêu cầu các chuyên gia duyệt xét lại những dự báo của họ trên xơ sở tóm tắt tất cả các dự báo có thể có những bổ sung thêm.

- Dự báo theo phương trình hồi quy: Theo phương pháp này, mức độ cần dự báo phải được xây dựng trên cơ sở xây dựng mô hình hồi quy, mô hình này được xây dựng phù hợp với đặc điểm và xu thế phát triển của hiện tượng nghiên cứu. Để xây dựng mô hình hồi quy, đòi hỏi phải có tài liệu về hiện tượng cần dự báo và các hiện tượng có liên quan. Loại dự báo này thường được sử dụng để dự báo trung hạn và dài hạn ở tầm vĩ mô.

- Dự báo dựa vào dãy số thời gian: Là dựa trên cơ sở dãy số thời gian phản ánh sự biến động của hiện tượng ở những thời gian đã qua để xác định mức độ của hiện tượng trong tương lai.

1.2 Dãy số thời gian

Khái niệm

Mặt lượng của hiện tượng thường xuyên biến động qua thời gian. Trong thống kê để nghiên cứu sự biến động này ta thường dựa vào dãy số thời gian. Dãy số thời gian là dãy số các trị số của chỉ tiêu thống kê được sắp xếp theo thứ tự thời gian. Ví dụ: có số liệu về doanh thu của Bưu điện X từ năm 1999 - 2003 như sau: ĐVT: tỷ đồng.

Năm	1999	2000	2001	2002	2003
Doanh thu	23,9	28,1	37,3	47,2	67,4

Bảng. 1.1

Ví dụ trên đây là một dãy số thời gian về chỉ tiêu doanh thu của đơn vị Bưu điện này từ năm 1999-2003. Qua dãy số thời gian có thể nghiên cứu các đặc điểm về sự biến động của hiện tượng, vạch rõ xu hướng và tính quy luật của sự phát triển, đồng thời để dự đoán các mức độ của hiện tượng trong tương lai. Mỗi dãy số thời gian có hai thành phần:

- Thời gian: có thể là ngày, tuần, tháng, quí, năm, ... Độ dài giữa hai thời gian liền nhau được gọi là khoảng cách thời gian.

- Chỉ tiêu về hiện tượng nghiên cứu: chỉ tiêu này có thể là số tuyệt đối, số tương đối, số bình quân. Trị số của chỉ tiêu còn gọi là mức độ của dãy số.

Phân loại dãy số thời gian:

Căn cứ vào tính chất thời gian của dãy số, có thể phân biệt thành 2 loại:

1. Dãy số thời kỳ: là dãy số biểu hiện mặt lượng của hiện tượng qua từng thời kỳ nhất định

2. Dãy số thời điểm: là loại dãy số biểu hiện mặt lượng của hiện tượng qua các thời điểm nhất định. Dãy số này còn được phân biệt thành 2 loại:

    - Dãy số thời điểm có khoảng cách thời gian đều nhau.

    - Dãy số thời điểm có khoảng cách thời gian không đều.

Các yếu tố ảnh hưởng đến biến động thời gian:

1. Biến động có xu hướng.

2. Biến động theo thời vụ.

3. Biến động theo chu kỳ.

4. Biến động bất thường.

1.3 Các bước xây dựng mô hình dự đoán

Giả sử với những dữ liêu sẵn có, ta có thể bắt đầu công việc xây dựng mô hình dự đoán qua các công đoạn sau:

1. Phân tích tổng thể dữ liệu
   Do sự tiến bộ về các công cụ và thuật toán machine learning, nên việc xây dựng mô hình dự đoán có thể làm rất nhanh và dễ dàng. Do đó, thay vì dành phần lớn thời gian để thiết kế lại những gì đã có sẵn (mô hình dự đoán), ta dành thời gian đó cho việc quan sát sơ bộ dữ liệu để đánh giá tổng thể về độ tin cây, nhận diện các dữ liệu còn thiếu, các dữ liệu biên, các trường dữ liệu không liên quan đễn vấn đề cần giải quyết. Thời gian nay sẽ giúp chúng ta hiểu rõ hơn dữ liệu mà mình đang làm việc, từ đó có cách tiếp cận đúng trong việc xây dựng mô hình dự đoán, tránh tình trạng mô hình được tạo ra dựa trên các giá trị không thực thế hay không tồn tại, ảnh hướng đến kết quả dự đoán.

2. Xử lý sơ bộ dữ liệu (Xử lý liệu biên, dữ liệu bị thiếu)
   Đây được xem là phần chiếm nhiều thời gian nhất, cần những biện pháp thông minh để hoàn tất giai đoạn này. Đây là những cách để chúng ta sử lý những dữ liệu xấu.

   · Gán những biến tạm cho các giá trị còn thiếu: các giá trị còn thiếu của một lượng thông tin có thể cho chúng ta biết nhiều điều. Bằng cách gán các giá trị tạm để mô hình dự đoán biết đó là giá trị còn thiếu có thể cho ra các kết quả chuẩn xác hơn.

   · Gán những giá trị còn thiếu bằng giá trị trung bình trong cùng một trường dữ liệu (data imputation). Đây cũng là cách phổ biến để xử lý các dữ liệu còn thiếu.

3. Xây dựng mô hình dự đoán
   Sử dụng các công cụ hoặc các thuật toán để xây dựng mô hình dựa trên các dữ liệu đã được xử lý.

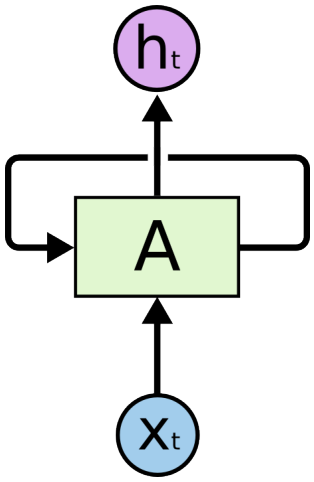
4. Đánh giá sự chính xác của mô hình.
   Đánh giá độ chính xác của các Model dựa các công thức đã sẵn có.

1.4 Recurrent neural network

Con người không bắt đầu suy nghĩ từ đầu mỗi giây. Khi bạn đọc bài luận này, bạn hiểu từng từ dựa trên sự hiểu biết của bạn về các từ trước đó. Bạn không nên ném mọi thứ đi và bắt đầu suy nghĩ lại từ đầu. Suy nghĩ của bạn có sự lưu lại.

Mạng lưới thần kinh truyền thống có thể làm được điều này, và nó có vẻ như là một thiếu sót lớn. Ví dụ, hãy tưởng tượng bạn muốn phân loại loại sự kiện nào đang diễn ra tại mọi thời điểm trong phim. Nó không rõ làm thế nào một mạng lưới thần kinh truyền thống có thể sử dụng lý lẽ của nó về các sự kiện trước đó trong phim để thông báo cho những sự kiện sau này.

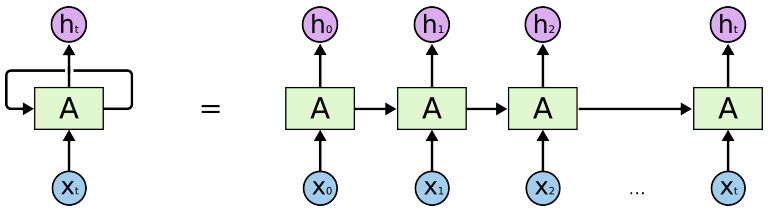
Recurrent neural network giải quyết vấn đề này. Chúng là các mạng có các vòng lặp trong đó, cho phép thông tin tồn tại.



Hình. 1.1: Recurrent Neural Network có vòng lặp.

Trong sơ đồ trên, một đoạn của mạng thần kinh, A, xem xét một số x_t đầu vào và xuất ra một giá trị h_t.Một vòng lặp cho phép thông tin được truyền từ một bước của mạng sang bước tiếp theo.

Những vòng lặp này làm cho Recurrent neural network có vẻ như bí ẩn. Tuy nhiên, nếu bạn suy nghĩ nhiều hơn một chút, hóa ra họ không phải là một mạng lưới thần kinh bình thường. Recurrent neural network có thể được coi là nhiều bản sao của cùng một mạng, mỗi bản tin truyền cho một người kế nhiệm. Xem xét những gì xảy ra nếu chúng ta bỏ vòng lặp:



Hình. 1.2: Recurrent Neural Network đã được trải ra.

Bản chất giống như chuỗi này cho thấy các Recurrent neural network có liên quan mật thiết đến các chuỗi và danh sách. Nó sử dụng kiến trúc tự nhiên của mạng neuron để sử dụng cho dữ liệu đó.

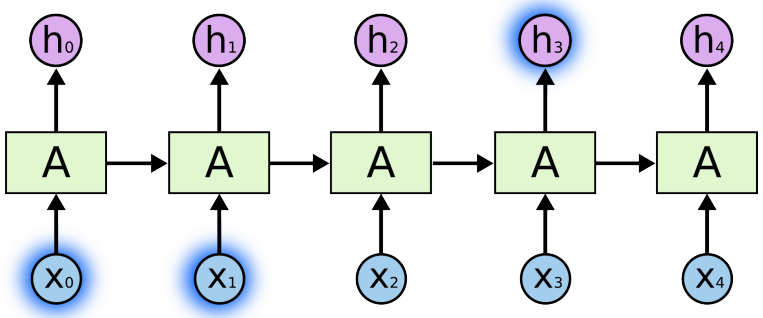
Và chúng chắc chắn được sử dụng! Trong vài năm qua, đã có những thành công đáng kinh ngạc khi áp dụng RNN cho nhiều vấn đề khác nhau: nhận dạng giọng nói, mô hình ngôn ngữ, dịch thuật, chú thích hình ảnh.

Điều cần thiết cho những thành công này là việc sử dụng "LSTM", một loại Recurrent neural network rất đặc biệt, hoạt động, cho nhiều tác vụ, tốt hơn nhiều so với phiên bản tiêu chuẩn. Hầu như tất cả các kết quả thú vị dựa trên Recurrent neural network đều đạt được với chúng. Nó có những LSTM mà bài tiểu luận này sẽ khám phá.

Vấn đề phụ thuộc xa

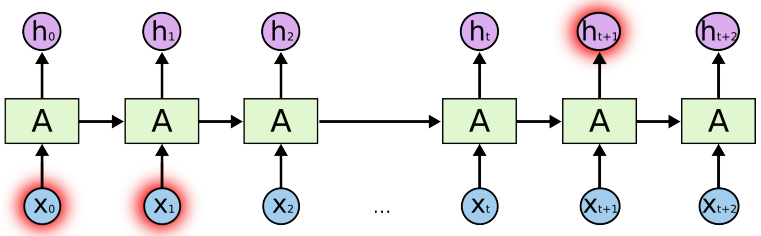
Một trong những lời kêu gọi của RNN là ý tưởng rằng họ có thể kết nối thông tin trước đó với tác vụ hiện tại, chẳng hạn như sử dụng các khung video trước đó có thể thông báo cho sự hiểu biết về khung hiện tại. Nếu RNN có thể làm điều này, thì họ cực kỳ hữu ích. Nhưng họ có thể? Không hẳn.

Đôi khi, chúng ta chỉ cần nhìn vào thông tin gần đây để thực hiện nhiệm vụ hiện tại. Ví dụ, hãy xem xét một mô hình ngôn ngữ đang cố gắng dự đoán từ tiếp theo dựa trên các từ trước đó. Nếu chúng ta đang cố gắng dự đoán từ cuối cùng trong "các đám mây trên *bầu trời*", thì chúng ta không cần bất kỳ bối cảnh nào nữa - đó là một điều khá rõ ràng, từ tiếp theo sẽ là *bầu trời*. Trong những trường hợp như vậy, khi khoảng cách giữa thông tin liên quan và địa điểm mà nó cần là nhỏ, RNN có thể học cách sử dụng thông tin trong quá khứ.



Nhưng cũng có những trường hợp chúng ta cần nhiều bối cảnh hơn. Cân nhắc việc cố gắng dự đoán từ cuối cùng trong văn bản. "Tôi lớn lên ở Việt Nam. Tôi nói tiếng trôi chảy tiếng *Việt*". Thông tin gần đây cho thấy từ tiếp theo có lẽ là tên của một ngôn ngữ, nhưng nếu chúng ta muốn thu hẹp ngôn ngữ nào, chúng ta cần thu hẹp ngôn ngữ nào bối cảnh của Việt Nam, từ phía trước. Nó hoàn toàn có thể cho khoảng cách giữa thông tin liên quan và điểm cần thiết để trở nên rất lớn.

Thật không may, khi khoảng cách đó tăng lên, các RNN trở nên không thể học cách kết nối thông tin.



Về lý thuyết, các RNN hoàn toàn có khả năng xử lý các phụ thuộc dài hạn như vậy. Một người có thể cẩn thận chọn các tham số cho họ để giải quyết các vấn đề theo hình thức này. Đáng buồn thay, trong thực tế, RNN trông dường như không có thể học chúng. Vấn đề đã được khám phá sâu bởi Hochreiter (1991) [German] và Bengio, et al. (1994), người đã tìm thấy một số lý do khá cơ bản tại sao nó có thể khó khăn.

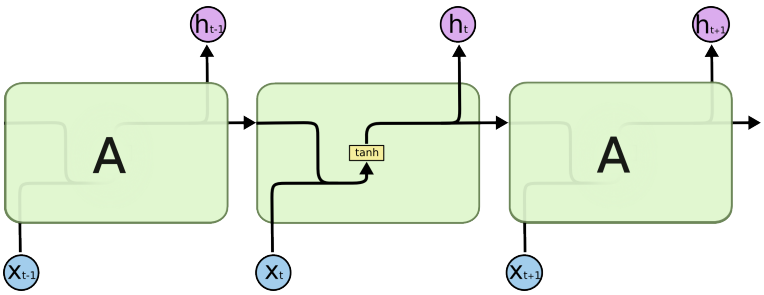
Rất may, LSTM không có vấn đề này!

Mạng LSTM

Long Short Term Memory (mạng bộ nhớ dài ngắn hạn) - thường được gọi là LSTM của - - là một loại RNN đặc biệt, có khả năng học các phụ thuộc xa. Chúng được giới thiệu bởi Hochreiter & Schmidhuber (1997), và được nhiều người tinh chỉnh và phổ biến. Chúng hoạt động rất tốt trong nhiều vấn đề lớn, và hiện đang được sử dụng rộng rãi.

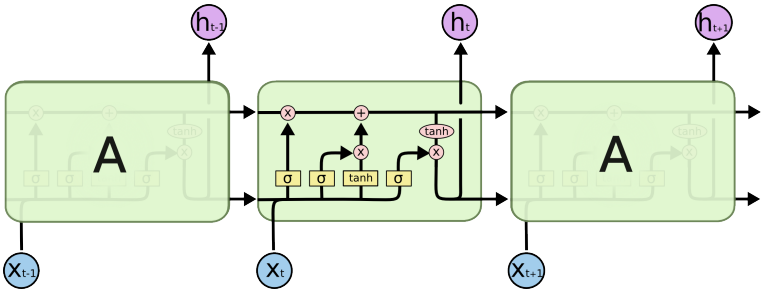
Các LSTM được thiết kế rõ ràng để tránh vấn đề phụ thuộc dài hạn. Ghi nhớ thông tin trong thời gian dài thực tế là hành vi mặc định của nó, không phải là thứ khó khăn để học!

Tất cả các mạng thần kinh tái phát có dạng một chuỗi các module lặp lại của mạng thần kinh. Trong các RNN tiêu chuẩn, module lặp lại này sẽ có cấu trúc rất đơn giản, chẳng hạn như một lớp $tanh$ duy nhất.

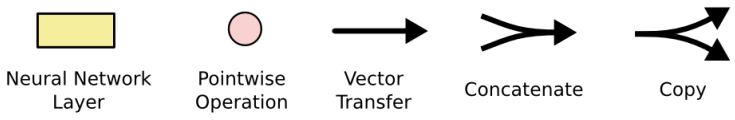


Hình. 1.3: Module lặp trong RNN chuẩn chứa một lớp duy nhất.

LSTM cũng có cấu trúc chuỗi, nhưng các module lặp có một cấu trúc khác. Thay vì có một lớp mạng thần kinh duy nhất,nó có bốn lớp, tương tác theo một cách rất đặc biệt.



Hình. 1.4: Module lặp trong LSTM chứa 4 lớp tương tác.



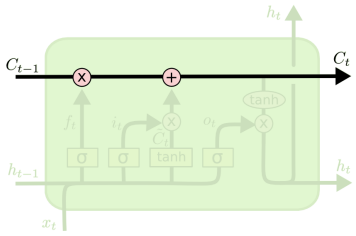
Hình. 1.5: Các ký hiệu trong LSTM.

Trong sơ đồ trên, mỗi dòng mang toàn bộ một vector, từ đầu ra của một nút đến đầu vào của các nút khác. Các vòng tròn màu hồng đại diện cho các phép toán, như phép cộng vector, trong khi các hình chữ nhật màu vàng biểu thị các mạng thần kinh để học. Các dòng hợp nhất biểu thị việc ghép nối, trong khi một dòng phân tách biểu thị nội dung của nó được sao chép và các bản sao đi đến các vị trí khác nhau.

Ý tưởng chính của LSTM

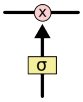
Ý tưởng chính của LSTM là ô trạng thái, đường ngang chạy qua đỉnh sơ đồ.

Dòng trạng thái giống như một băng chuyền. Nó chạy thẳng xuống toàn bộ chuỗi, chỉ với một số tương tác tuyến tính nhỏ dọc bên cạnh, dễ dàng cho thông tin truyền theo.



LSTM có khả năng loại bỏ hoặc thêm thông tin vào ô trạng thái, được điều chỉnh cẩn thận bởi các cấu trúc gọi là cổng.

Cổng là một cấu trúc điều khiển thông tin thông qua. Chúng được cấu tạo từ một lớp lưới thần kinh $sigmold$ và một phép toán nhân.



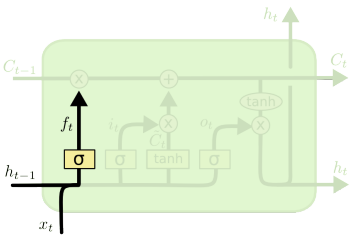
Đầu ra của các lớp $sigmoid$ có giá trị $[0, 1]$, mô tả mức độ cho qua. Giá trị bằng 0 có nghĩa là không để bất cứ thứ gì qua, trong khi giá trị của 1 nghĩa là có thể cho phép mọi thứ thông qua!

Một LSTM có ba trong cổng này, để bảo vệ và kiểm soát trạng thái tế bào.

Các bước LSTM hoạt động

Bước đầu tiên trong LSTM là quyết định thông tin nào đi ra khỏi ô trạng thái. Quyết định này được đưa ra bởi một lớp sigmoid được gọi là lớp "cổng quên". Nó dựa vào giá trị của h_{t-1} và x_t, và đưa ra một số từ 0 đến 1 tương ứng với mỗi số ô trạng thái c_{t-1}. Số 1 có nghĩa là giữ lại toàn bộ thông tin trong khi đó số 0 nghĩa là hãy quên nó đi.

Hãy xem ví dụ của chúng ta về một mô hình ngôn ngữ đang cố gắng dự đoán từ tiếp theo dựa trên tất cả các từ trước đó. Trong một vấn đề như vậy, ô trạng thái có thể bao gồm vai về của ngữ hiện tại, để có thể sử dụng các động từ một cách chính xác. Khi có một chủ ngữ mới, nó sẽ quên đi vai về của chủ ngữ cũ.

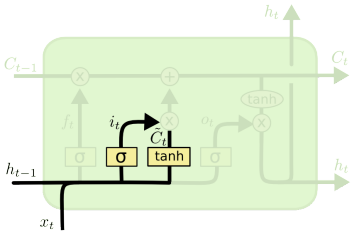


$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

Bước tiếp theo là quyết định những thông tin mới sẽ lưu trữ trong ô trạng thái. Việc này có hai phần. Đầu tiên, một lớp sigmoid được gọi là lớp "cổng đầu vào" quyết định giá trị nào sẽ cập nhật.

Tiếp theo, một lớp $tanh$ tạo ra một vector các giá trị ứng cử viên mới, C_t, có thể được thêm vào ô trạng thái. Sau đó sẽ kết hợp cả hai để tạo ra một bản cập nhật cho trạng thái.

Trong ví dụ về mô hình ngôn ngữ, chúng tôi muốn thêm vai vế của chủ ngữ mới vào ô trạng thái, để thay thế chủ ngữ đã quên.

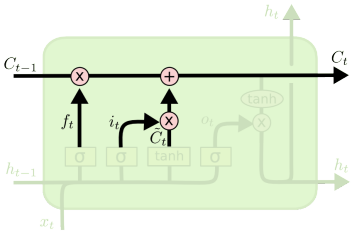


$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

Bây giờ, sẽ cập nhật ô trạng thái cũ, C_{t-1}, sang ô trạng thái mới C_t. Các bước trước đã quyết định phải quên và nhớ những gì, giờ là lúc thực hiện nó.

Nhân ô trạng thái cũ với f_t, quên đi những điều quyết định quên trước đó. Sau đó, cộng với $i_t * \tilde{C}_t$. Đây là giá trị ứng viên mới, được tính theo mức độ cập nhật từng giá trị trạng thái.

Trong trường hợp của mô hình ngôn ngữ, đây là lúc thực sự bỏ thông tin về vai vế và thêm thông tin mới, như đã quyết định trong các bước trước.



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

Cuối cùng là quyết định những gì sẽ xuất ra. Đầu ra sẽ được lọc dựa vào ô trạng thái. Đầu tiên, chạy một lớp $sigmoid$ quyết định phần nào của ô trạng thái mà sẽ xuất ra. Sau đó, đưa ô trạng thái qua hàm $tanh$ (để đẩy các giá trị nằm trong khoảng -1 đến 1) và nhân nó với đầu ra của cổng $sigmoid$, do đó chỉ đưa ra các dữ liệu đã quyết định

Đối với ví dụ về mô hình ngôn ngữ, vì nó chỉ nhìn thấy một chủ ngữ, nó có thể muốn đưa ra thông tin có liên quan đến một động từ, trong trường hợp đó là những gì sắp diễn ra. Ví dụ, nó có thể xuất ra vai vế của chủ ngữ , để biết được cách dùng từ với chủ ngữ đó.

2 Xây dựng mô hình

2.1 Giới thiệu

Dự đoán giá cổ phiếu là việc mà các nhà kinh tế thường xuyên bị hỏi. Tuy nhiên việc dự đoán chính xác là một việc khó có thể làm vì có rất nhiều yếu tố ảnh hưởng. Như cổ phiếu của một công ty chịu rất nhiều ảnh hưởng từ hoạt động công ty, chính phủ, quyết định của các nhà đầu tư. Một số cá nhân, tổ chức có ảnh hưởng cực kì lớn tới giá cổ phiếu. Với sự phát triển của internet thì thông tin được lan truyền rất nhanh, thông tin dường như gây ảnh hưởng đến giá cổ phiếu ngay lập tức

Đối với vàng thì thực sự không có một cá nhân tổ chức nào có thể nắm được quyền lực cụ thể. Có rất nhiều nhân tố, một dòng twitter của tổng thống Mỹ Donald Trump không thể nào có những ảnh hưởng đáng kể tới giá vàng như việc tuyên bố cấm cửa Huawei của ông.

Thời trước đã có nhiều phương pháp để dự đoán giá vàng bằng phương pháp xây dựng mô hình thuần thống kê như

- Autoregressive (AR)
- Moving Average (MA).
- Autoregressive Moving Average (ARMA)
- Autoregressive Integrated Moving Average (ARIMA)
- Seasonal Autoregressive Integrated Moving Average (SARIMA)

Những phương pháp này yêu cầu người dùng cần có rất nhiều kiến thức để tinh chỉnh tham số, đưa ra các dự đoán

Ngày nay, với sự phát triển của Machine Learning, có nhiều thuật toán mới tiến bộ, không yêu cầu người dùng phải có kiến thức rộng lớn như cây quyết định, học sâu.

Trong cây quyết định có thuật toán nổi tiếng như XGBoost, LightGBM. Những thuật toán yêu cầu người dùng có khả năng đưa ra nhiều feature nhất có thể để có thể đưa ra dự đoán chính xác

Học sâu có các mạng dòng họ Recurrent Neural Networks thường được sử dụng vì khả năng ghi nhớ thông tin trước đó. Phổ biến nhất là LSTM, GRU. Những thuật toán này không cần người dùng đưa ra những feature nhưng những mạng này thường yêu cầu dữ liệu rất nhiệu và sức mạnh tính toán rất lớn để đưa ra dự đoán chính xác.

Gần đây, Facebook có đưa ra thuật toán Prophet. Sử dụng các tham số mặc định, Prophet đưa ra các dự đoán chính xác đánh ngạc nhiên mà không tốn nhiều tài nguyên để tính toán.

2.2 Đặt vấn đề

Bài toán đặt ra cho biết thông tin giá vàng trên sàng chứng khoáng hiện tại (GLD ETF). Làm sao để dự đoán được giá vàng đóng phiên của 90 ngày tiếp theo.
Dự đoán được giá vàng sẽ giúp cho nhà đầu tư đưa ra được quyết định mua vào bán ra ngắn hạn, tạo điều kiện để tích luỹ số tiền. Còn việc dự đoán xa hơn 90 ngày thì rất khó có thể chính xác và ít có ý nghĩa. Như lời khuyên từ Jack Welch cựu CEO cựu General Electric, bạn nên để tất cả các trứng vào một giỏ và trông chừng nó. Hiếm có ai mà mua vàng để sinh lời mà lại không theo dõi giá cả hằng ngày.

2.3 Xây dựng mô hình giải quyết vấn đề

2.3.1 Dữ liệu

Được lấy từ package python fix_yahoo_finance.

```
import fix_yahoo_finance as yf
data = yf.download('GLD','2000-01-01')
data.to_csv('data/gld.csv')
```

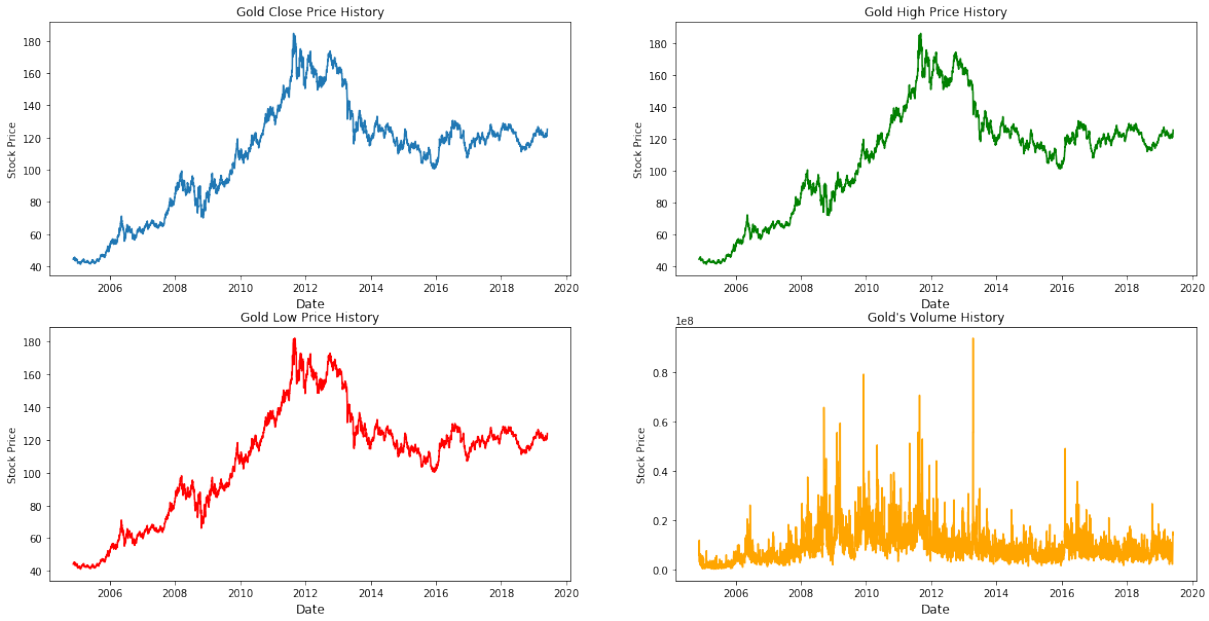
Gói này lấy thông tin từ trang Yahoo Finance và chuyển thành Panda Dataframe. Bảng này gồm 7 cột:

- Date: ngày
- Open: giá lúc mở phiên

- Close: giá lúc đóng phiên danh nghĩa
- High: giá cao nhất trong ngày
- Low: giá thấp nhất trong ngày
- Adj Close: giá lúc đóng phiên thực tế đã tính phần lạm phát
- Volume: số lượng bán ra

Date	Open	Close	High	Low	Adj Close	Volume
2004-11-18	44.43	44.490002	44.07	44.380001	44.380001	5992000

Bảng. 2.1



Hình. 2.1: Biểu đồ dữ liệu

Dữ liệu được lấy từ ngày **18-11-2004** đến **06-03-2019**. Dữ liệu được lấy từ thị trường chứng khoán, chính vì vậy, có những ngày nghỉ như ngày lễ, ngày thứ 7, chủ nhật sẽ không có dữ liệu. Có vẻ như dữ liệu không có tính thời vụ trong năm, tính thời vụ trong mùa và không bị ảnh hưởng bởi ngày lễ. Số lượng bán ra không phụ thuộc vào giá.

2.3.2 Xử lý dữ liệu

Đối với mạng học sâu

Chuyển thành bài toán học có giám sát

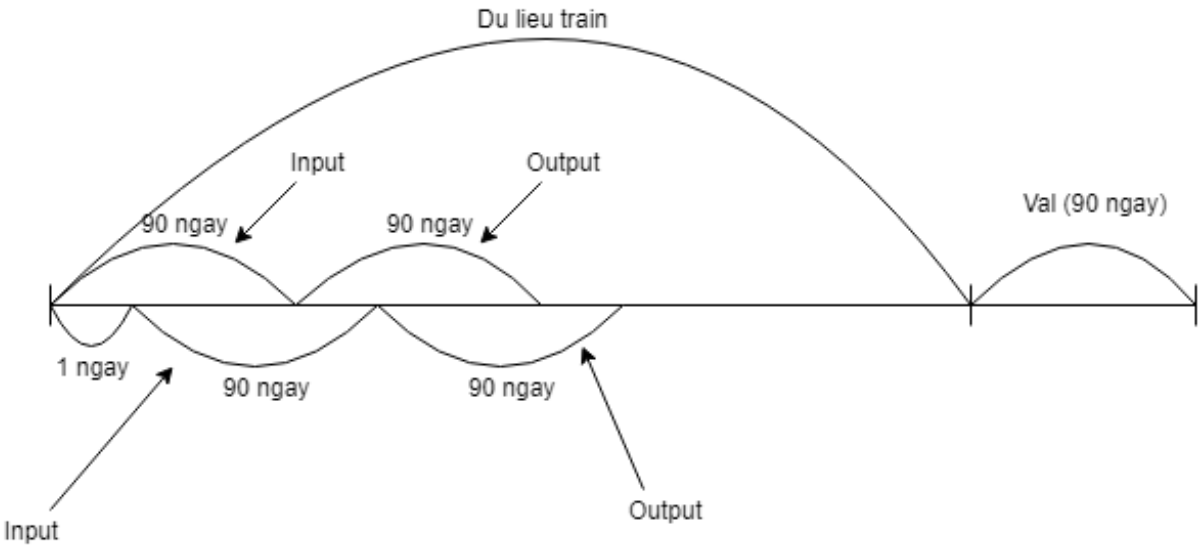
- Đối với những ngày không có dữ liệu thì điền dữ liệu bằng ngày trước để có tính liên tục

- Giữ lại cột: Date, Close. Đổi tên cột Date thành ds, Close thành y. Bỏ các cột còn lại(Theo quy chuẩn)

- Chuẩn hoá dữ liệu cột y bằng MinMaxScalar với khoảng (0, 1)

- Giữ 90 ngày cuối để xác minh. Những ngaỳ còn lại để đem train

- Đối với mỗi ngày, dữ liệu đầu vào: sẽ ra giá của 90 ngày trước ngày hiện tại, dữ liệu đầu ra là giá vàng của ngày hiện tại + 89 ngày sau ngày hiện tại. Vậy sẽ bỏ qua 90 ngày đầu của bộ dữ liệu (thiếu dữ liệu ngày trước đó). 90 ngày cuối vẫn giữ lại. Đối với những ngày không có dữ liệu thì điền vào 0



Hình. 2.2: Mô tả dữ liệu đầu vào đầu ra

y_past_1	y_past_2	y_past_3	...	y_past_88	y_past_89	y_past_90
0.009210	0.009000	0.005721	...	0.024559	0.024559	0.021768
...	...	...	...	...	...	...
0.512300	0.540010	0.531230	...	0.550120	0.550300	0.550309

Bảng. 2.2: Dữ liệu đầu vào mẫu

y	y_future_1	y_future_2	...	y_future_87	y_future_88	y_future_89
0.008791	0.010256	0.010396	...	0.004814	0.004814	0.004326
...	...	...	...	...	...	...
0.520001	0	0	...	0	0	0

Bảng. 2.3: Dữ liệu đầu ra mẫu

Đối với Prophet Chỉ cần đưa dữ liệu vào

- Đối với những ngày không có dữ liệu thì điền dữ liệu bằng ngày trước để có tính liên tục

- Giữ lại cột: Date, Close. Đổi tên cột Date thành ds, Close thành y. Bỏ các cột còn lại(Theo quy chuẩn)

- Giữ 90 ngày cuối để xác minh. Những ngày còn lại để đem train

ds	y
2019-03-01	121.879997
2019-03-02	121.879997
2019-03-03	121.823122
2019-03-04	121.842212
2019-03-05	121.851212

Bảng. 2.4: Dữ liệu mẫu Prophet

2.3.3 Xây dựng mô hình

Mạng học sâu

Mạng học sâu 1 là mạng LSTM chồng chất gồm 5 lớp: LSTM -> Dropout -> LSTM -> Dropout -> Dense

```
complex_model = Sequential()
complex_model.add(LSTM(units=100, input_shape=(X_train_vals.shape[1],
X_train_vals.shape[2]), return_sequences=True))
complex_model.add(Dropout(rate=0.2))
complex_model.add(LSTM(100, return_sequences=False))
complex_model.add(Dropout(rate=0.2))
complex_model.add(Dense(prediction_size, activation='linear'))
complex_model.compile(loss='mae', optimizer='adam')
```

Mạng học sâu 2 lúc sau đơn giản hơn là : LSTM -> Dense

```
basic_model = Sequential()
basic_model.add(LSTM(500, input_shape=(X_train_vals.shape[1],
X_train_vals.shape[2])))
```

```
basic_model.add(Dense(prediction_size))
basic_model.compile(loss='mae', optimizer='adam')
```

Đầu vào có dạng (90, 1)
Đầu ra có dạng (90) (dự đoán 90 ngày cùng lúc)
Mạng được train với loss function là $MAE = \frac{1}{n}\sum_{i=1}^{n}|p_i - a_i|$, optimizer là Adam, số epochs là 60

Prophet

Prophet với tham số mặc định, sử dụng daily seasonality, weekly seasonality, daily seasonality

```
m = Prophet(yearly_seasonality=True,
weekly_seasonality=True, daily_seasonality=True)
```

2.3.4 Phương pháp đánh giá

Trong bài toán dự đoán này, ta sử dụng **Walk Forward Validation**. Có một vài cần quyết định quyết định để đưa ra:

1. **Số lượng dữ liệu tối thiểu**: Đầu tiên, chúng ta phải chọn số lượng dữ liệu tối thiểu cần thiết để huấn luyện mô hình. Điều này có thể được coi là chiều rộng cửa sổ nếu sử dụng cửa sổ trượt.

2. **Cửa sổ trượt hoặc mở rộng**: Tiếp theo, chúng ta cần quyết định liệu mô hình sẽ được đào tạo trên tất cả dữ liệu mà nó có sẵn hay chỉ trên các dữ liệu gần đây nhất. Điều này xác định xem một cửa sổ trượt hoặc mở rộng sẽ được sử dụng.

Sau khi chọn cấu hình hợp lý, ở bài toán này là cửa sổ mở rộng, các mô hình có thể được đào tạo và đánh giá.

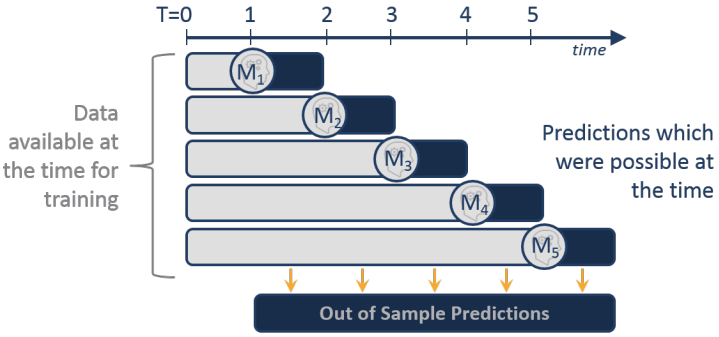
1. Bắt đầu từ đầu chuỗi thời gian, số lượng mẫu tối thiểu trong cửa sổ được sử dụng để huấn luyện một mô hình.

2. Mô hình đưa ra dự đoán cho bước tiếp theo.

3. Dự đoán được lưu trữ hoặc đánh giá theo giá trị đã biết.

4. Cửa sổ được mở rộng để bao gồm giá trị đã biết và quy trình được lặp lại (chuyển sang bước 1.)

Có thể hình dung như hình minh hoạ sau đây:



Hình. 2.3: Minh hoạ Walk Forward Validation

Sử dụng công thức MAE để đánh giá.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|p_i - a_i|$$

Trong đó

n là số ngày dự đoán

p_i là kết quả dự đoán vào ngày thứ i

a_i là kết quả thực tế vào ngày thứ i

Lấy 90 ngày cuối trong bộ dữ liệu để xác minh. Ta được kết quả như bảng sau:

Mô hình	MAE
Mạng học sâu 1 LSTM chồng chất	4.3483
Mạng học sâu 2 đơn giản	1.1131
Prophet	1.2269

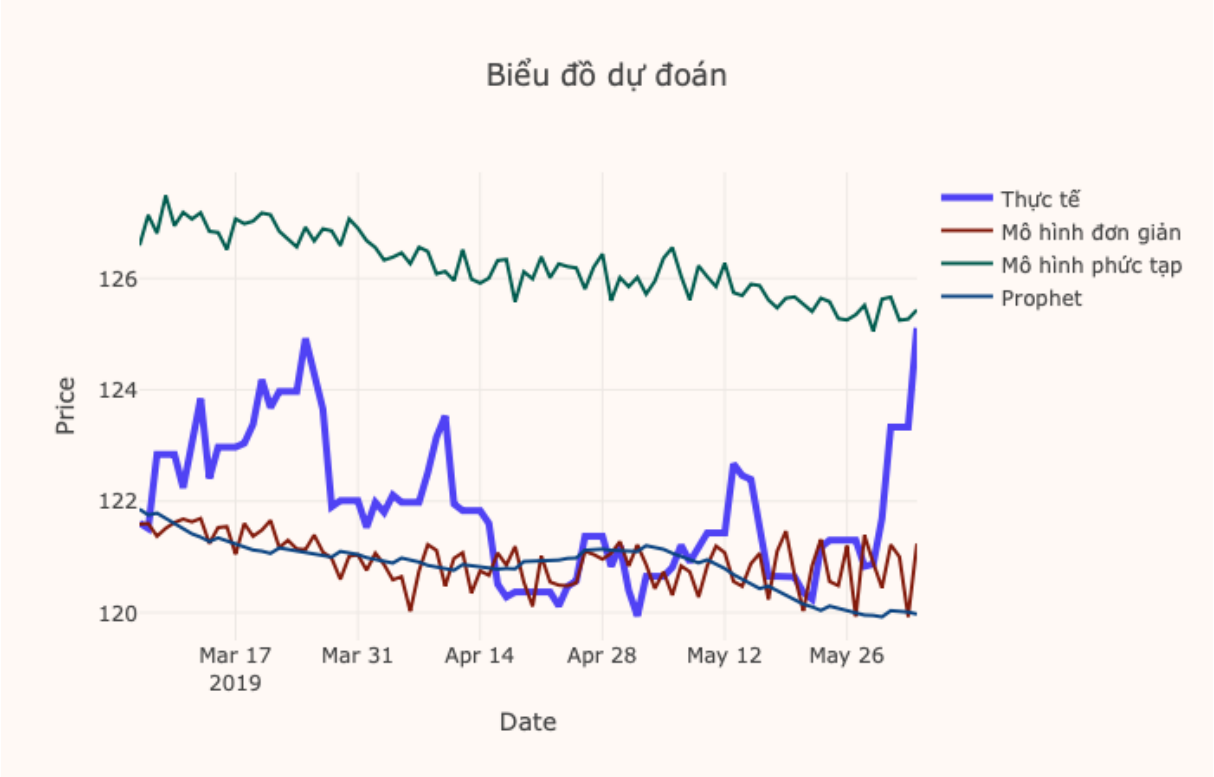
Bảng. 2.5: Kết quả

Tính trung bình MAE trong các bước **Walk Forward Validation**, với mỗi bước tăng thêm 360 mẫu dữ liệu để dự đoán 90 ngày sau. Ta được kết quả như bảng sau:

Mô hình	MAE
Mạng học sâu 1 LSTM chồng chất	17.1441
Mạng học sâu 2 đơn giản	16.0428
Prophet	5.9291

Bảng. 2.6: Kết quả **Walk Forward Validation**

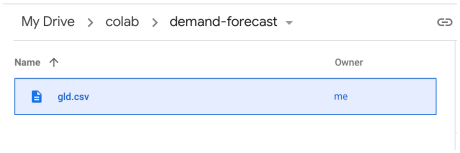
Prophet không tinh chỉnh nhiều tham số gì lại đưa ra kết quả chính xác hơn mạng LSTM chồng chất.



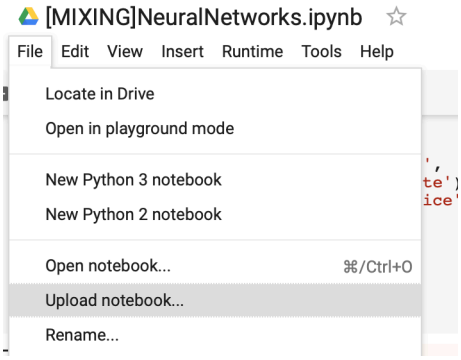
Hình. 2.4: Biểu đồ kết quả xác minh 90 ngày cuối

Hướng dẫn chạy

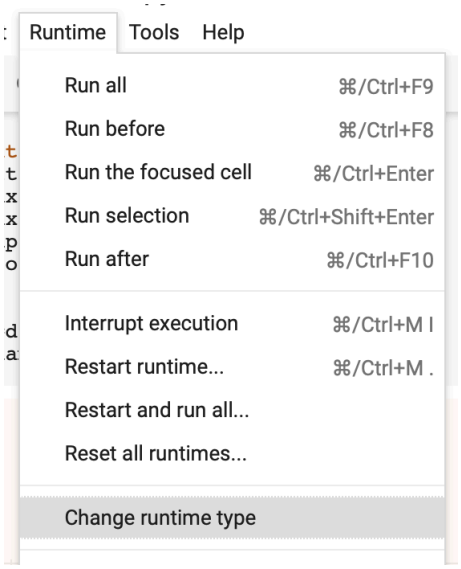
1. Vô Google Drive, tạo thư mục 'colab/demand-forecast' và up file 'gld.csv' trong thư mục 'data' lên

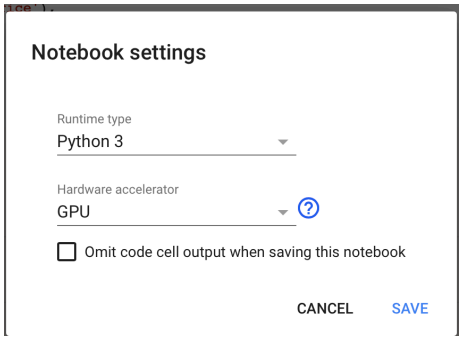


2. Vô Google Colab. Chọn File -> Upload notebook, up file Last90dayValidation.ipynb hoặc WalkForwaldValidation.ipynb lên Google Colab.

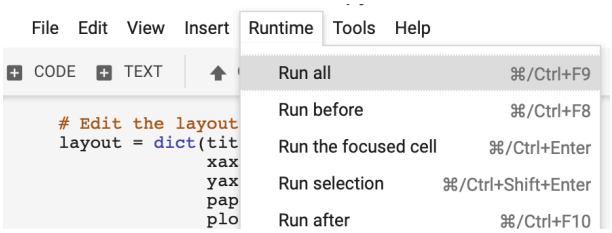


3. Chọn Runtime -> Change runtime type. Trong mục Hardware accelerator, chọn GPU. Nhấn Save.

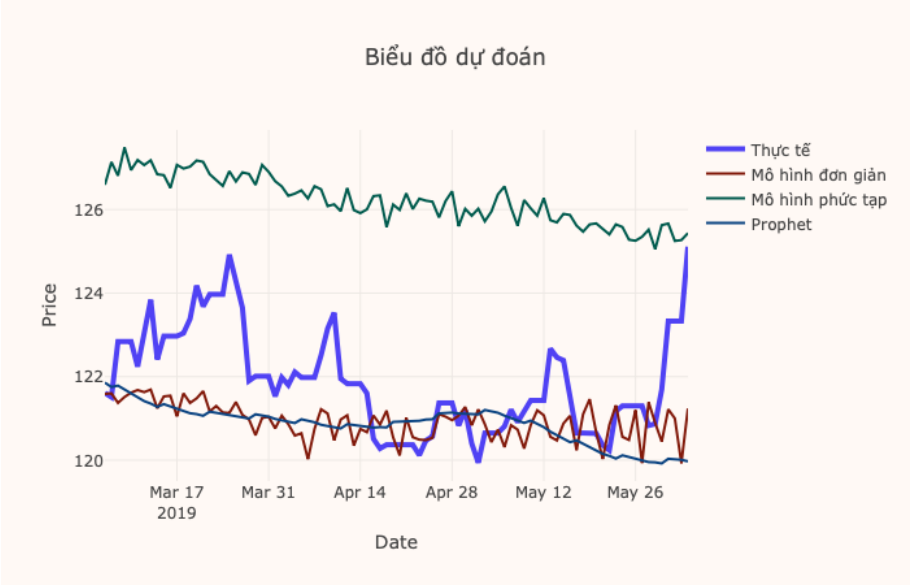

Runtime
Tools
Help
Run all ⌘/Ctrl+F9
Run before ⌘/Ctrl+F8
Run the focused cell ⌘/Ctrl+Enter
Run selection ⌘/Ctrl+Shift+Enter
Run after ⌘/Ctrl+F10
Interrupt execution ⌘/Ctrl+M I
Restart runtime... ⌘/Ctrl+M .
Restart and run all...
Reset all runtimes...
Change runtime type


Notebook settings
Runtime type
Python 3
Hardware accelerator
GPU
Omit code cell output when saving this notebook
CANCEL
SAVE

4. Chọn Runtime -> Run all để chạy. Lưu ý, trong lúc chạy sẽ thông báo truy cập vào link Google Drive để cấp quyền truy cập vào Google Drive ở cell thứ 1.



5. Kết quả



Tài liệu tham khảo

[Aun18] Jakob Aungiers. *Time series prediction using lstm deep neural networks*. 2018. URL: https://www.altumintelligence.com/articles/a/Time-Series-Prediction-Using-LSTM-Deep-Neural-Networks (visited on June 13, 2019).

[Bac19] Janio Martinez Bachmann. *S&P 500 Time Series Forecasting with Prophet*. 2019. URL: https://www.kaggle.com/janiobachmann/s-p-500-time-series-forecasting-with-prophet (visited on June 13, 2019).

[Hải17] Đỗ Minh Hải. *[RNN] Cài đặt GRU/LSTM*. 2017. URL: https://dominhhai.github.io/vi/2017/10/implement-gru-lstm/ (visited on June 13, 2019).

[Ngu09] Vũ Xuân Nam Nguyễn Thị Thanh Huyền Th.S Nguyễn Văn Huân. *Phân tích và dự báo kinh tế*. Trường Đại học Thái Nguyên, 2009.

[Ola15] Christopher Olah. *Understanding LSTM Networks*. 2015. URL: https://colah.github.io/posts/2015-08-Understanding-LSTMs/ (visited on June 13, 2019).