# Mô Hình DDPG trong Thư Viện FinRL Hỗ Trợ Giao Dịch Tài Chính Tự Động

Đỗ Vũ Gia Cần 19521271

***Tóm tắt nội dung*—Bài báo này sẽ trình bày về một thư viện học sâu tăng cường có tên là FinRL cùng với thuật toán đạo hàm chiến lược đơn định được cung cấp bởi thư viện và cách chúng dùng để hỗ trợ những nhà đầu tư xây dựng và phát triển chiến lược giao dịch chứng khoán của riêng họ một cách hiệu quả, một lĩnh vực đang được các nhà đầu tư đặc biệt quan tâm trong bối cảnh thị trường chứng khoán biến động rất phức tạp. Thực nghiệm được tiến hành bằng cách huấn luyện tác nhân thực hiện giao dịch trên những thị trường chứng khoán như DJIA, S&P 500, CSI 300, NASDAQ-100 và những đánh giá về chiến lược hoạt động của giao dịch sẽ được chỉ ra trong phần backtest.**

**Từ khóa: Học tăng cường sâu (gọi tắt là DRL), FinRL,Thuật toán đọa hàm chiến lược đơn định (gọi tắt là DDPG), mạng nơ-ron sâu (gọi tắt là mạng DNN) , DJIA, S&P 500, CSI 300, NASDAQ-100, backtest**

## I. GIỚI THIỆU

Giao dịch chứng khoán là một trong những phần quan trọng đối với lĩnh vực tài chính, công việc của nó về cơ bản là các nhà đầu tư sẽ phải quyết định xem nên đầu tư vào sàn chứng khoán nào với số lượng cổ phiếu bao nhiêu để tối ưu lợi nhuận thu về cho mình. Tuy nhiên những rủi ro liên quan đến lạm phát, lãi suất, hàng hóa, tính thanh khoản,... cùng với những biến động của thị trường chẳng hạn như đại dịch Covid đã tác động tiêu cực rất lớn đến thị trường tài chính làm cho việc giao dịch chứng khoán của các nhà đầu tư trở nên khó khăn và có những trường hợp dẫn đến lỗ nặng khi tham gia đầu tư, chính vì vậy việc đưa ra những chiến lược đầu tư hợp lý để để đối phó với những biến đổi phức tạp của thị trường chứng khoán đang rất được quan tâm bởi các nhà đầu tư lúc này.

Hiện nay, DRL[1,2] nổi lên như là một cách tiếp cận hiệu quả và được nhiều nhà nghiên cứu áp dụng để tìm ra những giải pháp cho bài toán tự động hóa giao dịch tài chính. Sở dĩ điều này là bởi vì mô hình mạng nơ ron sâu (deep neural network - mạng DNN) được áp dụng trong DRL có tác động mạnh mẽ trong việc xấp xỉ lợi nhuận dự kiến tại mỗi bước giao dịch đồng thời DRL kết hợp việc quan sát lợi nhuận hay rủi ro khi thực hiện giao dịch tiếp theo tại thời điểm giao dịch hiện tại (exploration) và những hành động giao dịch trong quá khứ (exploitation) để tìm ra chiến lược tối ưu. Trong bài báo này, chúng tôi sẽ trình bày về thuật toán DDPG[3,4] trong kỹ thuật DRL để mô hình hóa bài toán giao dịch chứng khoán tự động như là bài toán quy trình quyết định Markov (Markov Decision Process - MDP). Quá trình huấn luyện sẽ liên quan đến việc quan sát sự thay đổi giá chứng khoán, thực hiện giao dịch tại thời điểm tiếp theo và tính toán lợi nhuận để điều chỉnh chiến lược đầu tư phù hợp với mục đích là tối ưu lợi nhuận kì vọng.

Một trong những điểm nổi bật của thuật toán DDPG là khả năng hoạt động tốt trong không gian hành động liên tục. Điều này rất quan trọng bởi vì tác nhân giao dịch phải thực hiện quan sát trong một không gian trạng thái với nhiều sàn chứng khoán, mỗi sàn chứng khoán thì phải đưa ra quyết định là nên mua hoặc bán bao nhiêu cổ phiếu sao cho hợp lý làm cho không gian hành động rất lớn. Bên cạnh đó, thuật toán DDPG còn được thiết kế theo ba thành phần: (i) khung tiêu chuẩn actor-critic [5] bao gồm hai mạng là mạng Actor dùng phát sinh hành động và mạng Critic dùng để ước tính điểm thưởng kì vọng; (ii) mạng mục tiêu dùng để cập nhật giá trị hành động. Khi mạng Actor phát sinh được số hành động đủ lớn thì mạng mục tiêu này sẽ cập nhật giá trị của nó bằng cách lấy giá trị điểm thưởng kì vọng lớn nhất từ mạng critic, mục đích là để giữ sự ổn định cho quá trình huấn luyện; (iii) mô hình dịch chuyển (transition model) dùng cập nhật trải nghiệm.

Tiềm năng của kỹ thuật DRL, cụ thể là thuật toán DDPG trong giao dịch tài chính là rất lớn, tuy nhiên quá trình cài đặt và phát triển thuật toán phải trải qua những công việc phức tạp của bước tiêu chuẩn như quản lý các trạng thái giao dịch trung gian, chuẩn hóa số liệu đánh giá, rút trích những dữ liệu, thông số, tỷ lệ đặc trưng,... Những nhiệm này là rất khó khăn, dễ xảy ra sai sót trong quá trình thực hiện và tốn rất nhiều thời gian để xây dựng. Do đó, chúng tôi đã đề xuất thư viện FinRL[6,7] trong bài báo này để cải thiện cho tình trạng trên. Thư viện FinRL chứa những thuật toán tiêu chuẩn của DRL đã trải qua bước tinh chỉnh để phù hợp với thị trường chứng khoán đang thay đổi liên tục, trong đó bao gồm có thuật toán DDPG. Với FinRL, chúng tôi có thể cấu hình từ bộ dữ liệu thị trường chứng khoán trên môi trường ảo là các sàn chứng khoán như Dow Jones 30, DJIA, NASDAQ-100,... huấn luyện các tác nhân giao dịch như mạng DNN và đánh giá hoạt động giao dịch thông qua quá trình backtest[8]. Chúng tôi sẽ trình bày thư viện FinRL theo ba lớp: (i) lớp thấp nhất là môi trường giao dịch[9] dùng để mô phỏng thị trường chứng khoán cùng với các chỉ số tiêu chuẩn như giá cổ phiếu trong phiên giao dịch gần nhất (close), giá cổ phiếu ở phiên tiếp theo (open), khối lượng cổ phiếu có thể giao dịch (volume),... (ii) lớp tác nhân giao dịch sẽ trình bày về cách áp dụng thuật toán DDPG để huấn luyện tác nhân giao dịch; (iii) lớp ứng dụng (application layer) để áp dụng quy trình huấn luyện tác nhân cho giao dịch đơn cổ phiếu.

Phần còn lại của bài báo sẽ bao gồm: phần II tổng quan về các công trình nghiên cứu liên quan, phần III trình bày về thư viện FinRL và cách thuật toán DDPG được sử dụng, phần IV trình bày về những đánh giá thực nghiệm và phần V kết luận.

## II. CÁC CÔNG TRÌNH NGHIÊN CỨU LIÊN QUAN

Trong phần này, chúng tôi sẽ tổng quan qua về một số thư viện lập trình DRL được đóng gói (framework DRL) tương tự như thư viện FinRL và những thuật toán hiện đại thuộc kỹ thuật DRL.

### *A. Các thư viện lập trình DRL thông dụng*

Hiện nay, các nhà nghiên cứu đã phát triển những thư viện hỗ trợ cho máy học bằng python, điều này đã làm cho python trở thành một công cụ mạnh mẽ trong việc giải quyết các bài toán phức tạp mà con người không thể giải một cách chính xác, cũng như dự đoán, huấn luyện các mô hình. Và lĩnh vực định lượng tài chính cũng được hưởng lợi rất nhiều từ những phát triển này. Có rất nhiều những thư viện học tăng cường sâu có đặc tính tương tự như thư viện FinRL và dưới đây sẽ là một vài ví dụ:

- **OpenAI Gym** [10] là bộ công cụ phổ biến được xây dựng dựa trên những thuật toán học tăng cường sâu, cung cấp các môi trường tác vụ được chuẩn hóa, còn được gọi là môi trường gym và thực hiện huấn luyện đối tượng trên các môi trường này.
- **DeeR** [11] cũng là thư viện python dành cho học tăng cường sâu, được xây dựng với tính mô-đun để có thể dễ dàng đáp ứng các vấn đề khác nhau.
- **Tensorforce** [12] là một thư viện lập trình mã nguồn mở dành cho học tăng cường sâu, được sử dụng nhiều trong các lĩnh vực nghiên cứu. Nó có thiết kế dựa trên những thành phần mô-đun, tách biệt những thuật toán học sâu với những ứng dụng và có đầy đủ các mô hình TensorFlow[13].

### *B. Những thuật toán trong kỹ thuật DRL*

Trong kỹ thuật DRL, chúng ta thường có hai cách tiếp cận chính đó là (i) học dựa trên giá trị hành động và (ii) học dựa trên chiến lược.

- Đối với học dựa trên giá trị hành động, chiến lược tối ưu sẽ phụ thuộc vào việc lựa chọn thực hiện hành động có giá trị lớn nhất tại mỗi trạng thái. Giá trị hành động ở đây được hiểu là điểm thưởng kỳ vọng nhận được khi từ một trạng thái thực hiện một hành động để phát sinh ra trạng thái tiếp theo. Thuật toán tiêu biểu cho hướng tiếp cận này là mạng Q sâu (deep Q network - DQN), bên cạnh đó còn có các thuật toán như A2C [14], xấp xỉ chiến lược tối ưu (proximal policy optimization - PPO),...Đây là những thuật toán hoạt động dựa trên không gian hành động rời rạc.
- Học dựa trên chiến lược là cách tác nhân tại mỗi trạng thái sẽ lựa chọn hành động có xác suất lớn nhất để thực hiện, vì đặc trưng của quá trình học này là dựa trên không gian hành động rời rạc tức là không gian hành động rất lớn nên chúng ta có thể sử dụng một hàm mật độ xác suất để ước lượng xác suất cho mỗi hành động. Một số thuật toán áp dụng hướng tiếp cận này có thể kể đến như SAC [15], đạo hàm chiến lược đơn định (DDPG), đạo hàm chiến lược đơn định đa tác nhân [16]

Như đã giới thiệu ở trên thì để giải quyết bài toán tự động hóa giao dịch chứng khoán thì chúng ta phải huấn luyện tác nhân giao dịch trên nhiều sàn chứng khoán và mỗi sàn chứng khoán tác nhân đó phải quyết định xem nên mua hay bán bao nhiêu cổ phiếu làm cho không gian hành động trở nên rất lớn. Do đó các thuật toán theo hướng tiếp cận học dựa trên chiến lược sẽ là hợp lý hơn trong trường hợp này. Và chúng tôi chọn thuật toán DDPG vì tính phổ biến của nó cũng như tính đơn giản giúp nhà đầu tư dễ dàng tập trung nhiều hơn vào chiến lược giao dịch chứng khoán.

## III. THƯ VIỆN FINRL

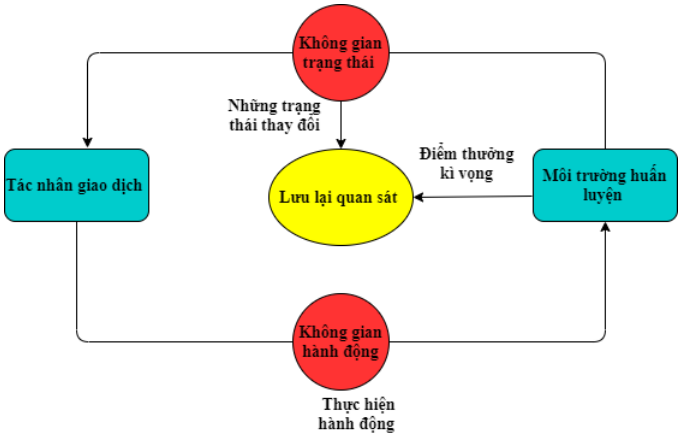
Thư viện FinRL sẽ được trình bày theo mô hình ba lớp: (i) lớp môi trường; (ii) lớp tác nhân ; (iii)lớp ứng dụng. Trong đó, trọng tâm mà chúng tôi hướng đến sẽ là ở lớp tác nhân, nơi chúng tôi sẽ trình bày về cách áp dụng thuật toán DDPG để huấn luyện tác nhân thực hiện giao dịch.

### *A. Môi trường huấn luyện tác nhân*

Khi nói về giao dịch chứng khoán thì giá trị cổ phiếu là thứ đáng được quan tâm nhất đối với những nhà đầu tư vì nó có tác động đến thành bại đối với chiến lược đầu tư của họ. Tuy nhiên, trên một thị trường chứng khoán thì có rất nhiều yếu tố ảnh hưởng đến giá trị cổ phiếu chẳng hạn như tương quan nghịch giữa thị trường chứng khoán với lạm phát, giá trị lãi suất tăng, những quy định của chính phủ về chính sách hàng hóa,...Do đó môi trường huấn luyện của chúng ta sẽ là môi trường ngẫu nhiên ( hay còn gọi là môi trường stochastic). Để kỹ thuật DRL hoạt động tốt trên môi trường này thì chúng tôi đề xuất giải pháp là mô hình hóa vấn đề về quy trình quyết định Markov, trong đó bao gồm các thành phần như sau:

- **Không gian trạng thái S:** mô tả những quan sát mà tác nhân nhận được từ môi trường như giá trị cổ phiếu, tổng tài sản hiện có, lợi nhuận kì vọng thu về, số lượng cổ phiếu cho phép giao dịch trên một sàn chứng khoán,...Được mô phỏng giống như một người giao dịch cần nắm bắt những thông tin trước khi thực hiện quyết định giao dịch.
- **Không gian hành động A:** mô tả những hành động cho tác nhân giống như những hành vi của người giao dịch là giữ lại hoặc mua, bán bao nhiêu lượng cổ phiếu. Chúng tôi sử dụng biến k để biểu thị cho các trường hợp: (i) k $\in (0; +\infty )$ : bán k cổ phiếu; (ii) k = 0 : giữ lại số cổ phiếu hiện có; (iii) k $\in (-\infty ; 0)$ : mua k cổ phiếu.
- **Điểm thưởng R(s, a, s'):** điểm thưởng nhận được khi thực hiện hành động a tại trạng thái s để mở ra một trạng thái mới s'.
- **Mô hình chuyển đổi T(s, a, s', r):** mô tả lại quá trình khi tác nhân thực hiện một hành động thì sẽ mở ra trạng thái mới như thế nào, nhận được điểm thưởng ra sao.
- **Hàm giá trị hành động Q(s, a):** mô tả giá trị của hành động a tại trạng thái s. Giá trị của hành động ở đây được hiểu là điểm thưởng kỳ vọng nhận được ở trạng thái tiếp theo khi thực hiện hành động a.
- **Chiến lược $\pi$:** mô tả chiến lược giao dịch tại trạng thái s. Thông thường, nó sẽ chứa một hàm phân phối xác suất

để thể hiện xác suất xảy ra của tất cả hành động có thể thực hiện được tại trạng thái s và quyết định xem nên thực hiện hành động nào.




Hình 1. Hoạt động của tác nhân giao dịch chứng khoán.

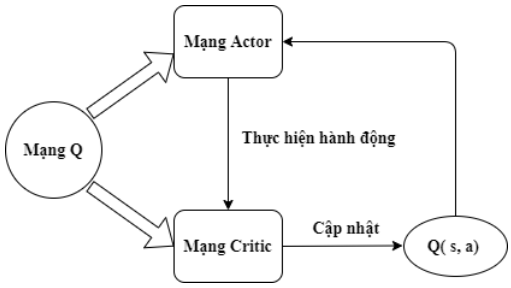
Nhiệm vụ của DRL là huấn luyện tác nhân giao dịch thực hiện các hành động để tương tác với môi trường để khám phá ra những trạng thái mới và điểm thưởng kỳ vọng tại trạng thái đó, kết hợp việc khai thác thông tin đã học từ những trạng thái cũ để thu được chiến lược tối ưu. Hoạt động của DRL được chỉ ra như trong Hình 1.
Trong quá trình thử nghiệm, chúng tôi sẽ dựa vào thư viện FinRL để mô phỏng môi trường giao dịch cho tác nhân vì thư viện này có chứa dữ liệu thông tin tiêu chuẩn của những thị trường chứng khoán thực tế như NASDAQ-100. DJIA, CSI 300, S&P 500.

### *B. Áp dụng thuật toán DDPG huấn luyện tác nhân giao dịch*

Như đã đề cập ở trên, tiếp theo chúng tôi sẽ đi đến đến lớp tác nhân của thư viện FinRL, nơi chúng tôi lựa chọn thuật toán DDPG để huấn luyện tác nhân tương tác với môi trường và chúng tối sẽ gọi tác nhân này là "tác nhân DDPG".

*a) Những yếu tố cần thiết::* Đầu tiên, chúng tôi cần xác định một số yếu tố cần thiết để xây dựng thuật toán DDPG:

- Các thời điểm giao dịch: Chúng tôi quy ước $s_0$ là trạng thái bắt đầu, tương ứng với ngày nhà đầu tư bắt đầu tham gia giao dịch; $s_t$ là trạng thái hiện tại, tương ứng với phiên giao dịch hiện tại của nhà đầu tư; $s_{t+1}$ là trạng thái tiếp theo tương ứng với phiên giao dịch tiếp theo; $s_f$ là trạng thái kết thúc tương ứng với ngày nhà đầu tư muốn kết thúc giao dịch.
- Mạng Q: mục đích là để cập nhật hàm giá trị hành động Q(s, a). Trong mạng này chúng tối thiết kế thêm hai mạng con có tên là mạng actor và mạng critic.
- Mạng actor: chứa thông tin trạng thái và giá trị của những hành động khả thi tại trạng thái đó. Chúng tôi đặt $\theta$ là tham số cho mạng này để tiện cho việc cập nhật.
- Mạng critic: dùng để cập nhật giá trị Q(s, a) sau khi hành động tối ưu được thực hiện. Chúng tôi đặt $\phi$ là tham số cho mạng này để tiện cho việc cập nhật.
- Mạng mục tiêu: là bản sao của mạng Q, mục đích là để khi mạng actor trải nghiệm số hành động đủ nhiều tại một trạng thái và chọn được hành động tối ưu thì chúng ta sẽ cập nhật mạng này để lưu trữ thông tin tại trạng thái đó. Chúng tôi cũng sẽ đặt $\theta_{mt}$ và $\phi_{mt}$ là những tham số của mạng này để tiện cho việc cập nhật.
- Bộ lưu trữ $\Re$: bộ phát lại trải nghiệm dùng để lưu trữ những dịch chuyển của tác nhân DDPG từ trạng thái này sang trạng thái khác.




Hình 2. Mô hình mạng Q.

*b) Các bước hoạt động: :* Cũng giống như những thuật toán DRL khác, tác nhân DDPG cũng phải trải qua các bước đó là (i) lựa chọn thực hiện hành động tại trạng mỗi trạng thái dựa trên những thứ đã biết; (ii) quan sát điểm thưởng r và trạng thái tiếp theo; (iii) dựa vào những quan sát để cải thiện những hiểu biết và tìm ra chiến lược tối ưu. Đây là một vấn đề tương đối khó, bởi vì ban đầu tác nhân không có thông tin về những trạng thái hay những thuận lợi hoặc rủi ro ở phía trước mà chính nó phải tự học thông qua việc tương tác đối với môi trường.
Để huấn luyện tác nhân DDPG giải quyết vấn đề khó khăn này, đầu tiên chúng tôi cần phải khởi tạo những yếu tố cần thiết đã được nêu ở trên bao gồm: thông tin trạng thái bắt đầu ($s_0$) và trạng thái kết thúc ($s_f$), các tham số của mạng Q ($\theta$, $\phi$) và mạng mục tiêu ($\theta_{mt}$, $\phi_{mt}$), giá trị Q(s, a) = 0, bộ lưu trữ $\Re$ rỗng.
Sau khi đã khởi tạo những yếu tố cần thiết, chúng tôi tiến hành cho tác nhân DDPG lặp lại công việc sau :

- Tại mỗi trạng thái $s_t$, lựa chọn thực hiện ngẫu nhiên một hành động $a_t$ để khám phá ra những trạng thái mới. Mỗi lần thực hiện hành động mới, ta lưu lại trải nghiệm của hành động đó bao gồm các thông tin ($s_t$, $a_t$, $s_{t+1}$, r) vào trong bộ lưu lưu trữ $\Re$.
- Khi thời gian thực hiện hành động ngẫu nhiên đạt đến ngưỡng $\epsilon$ cho trước, chúng ta tiến hành tính giá trị hành động cho hàm mục tiêu theo công thức:

$$y = r(s_t, a_t, a_{t+1}) + \gamma Q_{\phi_{mt}}(s_{t+1}, \mu_{\theta_{mt}}(s_{t+1})) \quad (1)$$

Trong đó:
$\gamma$ là hệ số chiết khấu lợi nhuận, $\gamma \in (0;1)$
$\mu_{\theta_{mt}}(s_{t+1})$ giá trị trung bình của mạng mục tiêu.

- Tiếp theo chúng ta cập nhật mạng Critic bằng cách sử dụng đạo hàm:

$$\nabla_\phi \frac{1}{N} \sum_{(s_t,a_t,r,a_{t+1})\in N} (Q_\phi(s_t,a_t)-y)^2 \quad (2)$$

với N là số hành động mà tác nhân đã khám phá được tại trạng thái s có trong $\Re$.
Mục đích của việc cập nhật mạng Critic là để giảm thiểu độ chênh lệch kỳ vọng giữa mạng Q và mạng mục tiêu.
- Cập nhật Actor để tìm chiến lược tối ưu bằng cách sử dụng đạo hàm:

$$\nabla_\phi \frac{1}{N} \sum_{s_t\in N} Q_\phi(s_t,\mu_\theta(s_t)) \quad (3)$$

Thông thường thì tác nhân DDPG sẽ tính toán điểm thưởng tại trạng thái $s_t$ theo phương trình Bellman:

$$Q(s_t,a_t) = E_{s_{t+1}}[r(s_t,a_t,s_{t+1}) + \gamma \max_{a_{t+1}} Q(s_{t+1}a_{t+1})] \quad (4)$$

- Sau khi đã cập nhật mạng Critic và mạng Actor ta tiến hành cập nhật mạng mục tiêu dựa theo những hiểu biết đã có của tác nhân được lưu trong bộ lưu trữ. Mạng mục tiêu được cập nhật theo công thức dưới đây:

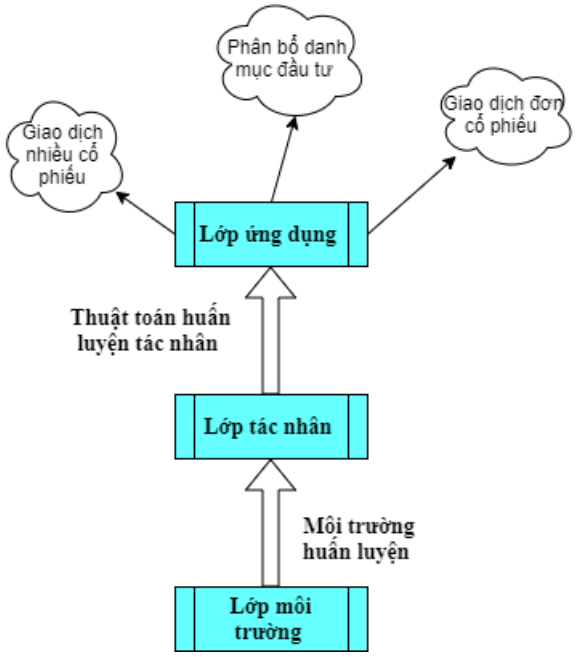
$$\theta_{mt} \longleftarrow \tau\theta_{mt} + (1-\tau)\theta \quad (5)$$

$$\phi_{mt} \longleftarrow \tau\phi_{mt} + (1-\tau)\phi \quad (6)$$

Tác nhân DDPG sẽ lặp lại các bước hoạt động như trên cho đến khi gặp trạng thái kết thúc và sẽ trả về tổng điểm thưởng $\sum_{t=1}^{t=f-1} r(s_t, a_t, s_{t+1})$ tối ưu cùng với chiến lược giao dịch tại mỗi thời điểm.
Tuy nhiên như đã đề cập ở trên thì việc cài đặt thuật toán này rất khó khăn và dễ xảy ra lỗi trong quá trình xây dựng, cho nên chúng tôi sẽ sử dụng mô hình DDPG có sẵn trong thư viện FinRL để mô phỏng công việc huấn luyện tác nhân.

### C. Những ứng dụng trong thực tế

Như chúng ta đã biết thì đầu tư chứng khoán là lĩnh vực hết sức phức tạp và tiềm ẩn nhiều rủi ro, tuy nhiên nếu gặp thời kỳ thuận lợi và có chiến lược đầu tư hợp lý thì nhà đầu tư cũng có thể thu về cho mình lợi nhuận to lớn mà không tốn nhiều công sức lao động. Với những thách thức và cơ hội như vậy, chủ đề trí tuệ nhân tạo áp dụng cho giao dịch chứng khoán tự động, nơi mà các nhà nghiên cứu xây dựng và huấn luyện các tác nhân để tham gia giao dịch với mục tiêu cắt giảm bớt rủi ro thua lỗ và tối ưu hóa lợi nhuận cho đầu tư đang rất được quan tâm lúc này.
Những nhà đầu tư có thể dựa vào kết quả huấn luyện tác nhân giao dịch trên các thị trường chứng khoán thực tế để có được thông tin về những bước giao dịch trong quá khứ , cũng như xem xét những biến động của thị trường chứng khoán đó trước đây để từ đó hình thành nên những ý tưởng đầu tư, đối phó được với sự lên xuống của thị trường.
Có ba lĩnh vực mà nhà đầu tư có thể tham khảo vào tác nhân giao dịch bao gồm: (i) giao dịch đơn cổ phiếu; (ii) giao dịch nhiều cổ phiếu và (iii) phân bổ danh mục đầu tư.




Hình 3. Sơ đồ mô hình 3 lớp của FinRL.

## IV. THỰC NGHIỆM

Quá trình thực nghiệm được tiến hành dựa trên những môi trường thị trường chứng khoán thực tế được cung cấp bởi thư viện FinRL bao gồm NASDAQ-100; DJIA; S&P 500; SSE 50; CSI 300; HSI. Môi trường ban đầu chứa các thông tin như: ngày giao dịch (date), tên các sàn chứng khoán (tic), giá mở cửa phiên giao dịch (open), giá đóng cửa phiên giao dịch (close), giá cổ phiếu cao nhất (high), giá cổ phiếu thấp nhất (low), số lượng cổ phiếu cho phép giao dịch (volume) và các tỉ số tài chính liên quan. Những thông tin ban đầu như trong Hình 4.

| | date | tic | open | high | low | close | volume | OPM | NPM | ROA |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 2009-01-02 | AAPL | 3.067143 | 3.251429 | 3.041429 | 2.787006 | 746015200.0 | 0.217886 | 0.163846 | 0.103222 |
| 0 | 2009-01-02 | AXP | 18.570000 | 19.520000 | 18.400000 | 15.657365 | 10955700.0 | 0.093973 | 0.072040 | 0.014094 |
| 0 | 2009-01-02 | BA | 42.799999 | 45.560001 | 42.779999 | 33.941101 | 7010200.0 | 0.047307 | 0.032525 | 0.026400 |
| 0 | 2009-01-02 | CAT | 44.910000 | 46.980000 | 44.709999 | 32.830360 | 7117200.0 | 0.124545 | 0.066662 | 0.040891 |
| 0 | 2009-01-02 | CSCO | 16.410000 | 17.000000 | 16.250000 | 12.505757 | 40980600.0 | 0.234698 | 0.196418 | 0.097593 |

Hình 4. Các thông tin ban đầu.

### A. Các tỷ lệ đánh giá

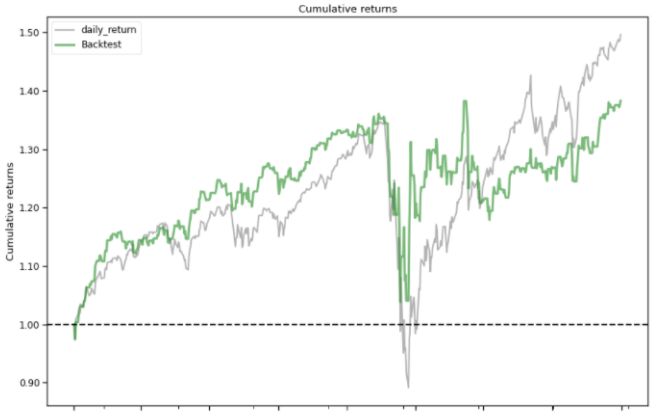
Các tỷ lệ đánh giá cơ bản được cung cấp để giúp chúng ta đánh giá hoạt động của chiến lược giao dịch bao gồm: lợi nhuận hằng năm (Annual return), lợi nhuận tích lũy (Cumulative returns), biến động hàng năm (Annual volatility), tỷ lệ Sharpe [18] (Sharpe ratio), tỷ lệ sụt giảm của tài khoản (Max drawdown).

| | Backtest |
|---|---|
| Annual return | 8.573% |
| Cumulative returns | 17.92% |
| Annual volatility | 26.641% |
| Sharpe ratio | 0.44 |
| Calmar ratio | 0.31 |
| Stability | 0.02 |
| Max drawdown | -27.724% |
| Omega ratio | 1.11 |
| Sortino ratio | 0.63 |
| Skew | 0.31 |
| Kurtosis | 17.53 |
| Tail ratio | 0.85 |
| Daily value at risk | -3.31% |
| Alpha | -0.07 |
| Beta | 0.80 |

Hình 5. Các số liệu đánh giá chiến lược.

Hình 5 biểu diễn các tỷ lệ để nhà đầu tư đánh giá tổng quan về hoạt động giao dịch

*B. Kết quả thực nghiệm*

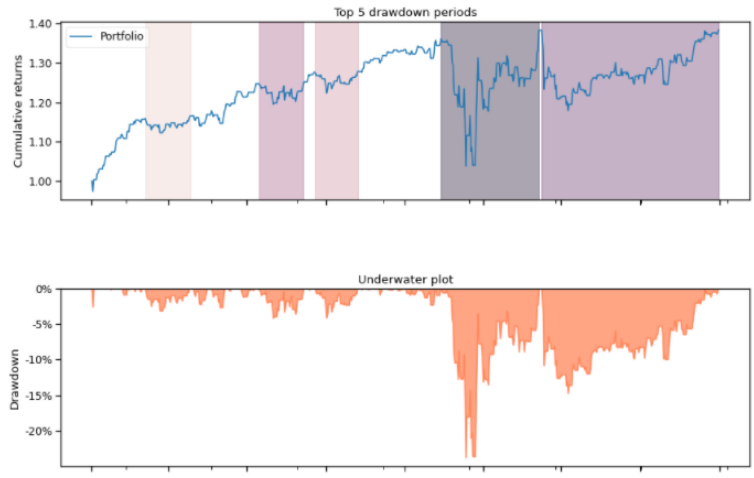



Hình 6. Lợi nhuận hàng năm.

Để mô phỏng tốt hơn giao dịch thực tế, chúng tôi sử dụng công cụ để đánh giá hoạt động giao dịch. Trong phần mô tả thực nghiệm này, chúng tôi sẽ chọn thị trường chứng khoán S&P 500 để làm ví dụ minh họa, với thời gian giao dịch là từ ngày 1/1/2019- 31/12/2020.
Hình 6 biểu diễn lợi nhuận hằng năm

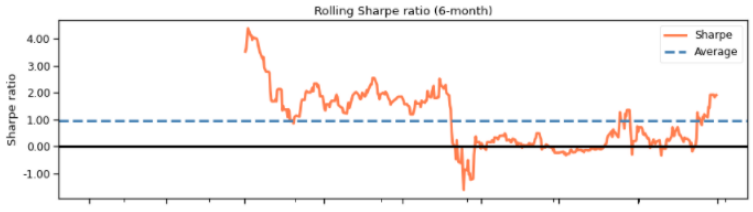

Hình 7. Biểu đồ lợi nhuận.

Hình 7 biểu thị lợi nhuận hằng năm và hằng tháng




Hình 8. Mức sụt giảm.

Hình 8 cho thấy tỷ lệ sụt giảm của tài khoản qua các giai đoạn




Hình 9. Tỷ lệ Sharpe.

Hình 9 biểu thị tỷ lệ Sharpe

## V. KẾT LUẬN

Trong bài báo này, chúng tôi đã trình bày về mô hình ba lớp của thư viện FinRL và tận dụng những tài nguyên sẵn có trong

thư viện để tiến hành thực nghiệm. Bên cạnh đó, chúng tôi còn trình bày về thuật toán DDPG trong mô hình DDPG của thư viện, cách huấn luyện một tác nhân DDPG thực hiện giao dịch chứng khoán tự động. Chúng tôi cũng đã mô phỏng việc huấn luyện tác nhân thực hiện giao dịch đa cổ phiếu trên thị trường chứng khoán S&P 500. Kết quả chiến lược mô phỏng thực nghiệm có thể tốt trên thị trường này nhưng cũng có thể lại kém hiệu quả trên thị trường khác.

Trong tương lai, chúng tôi sẽ cố gắng cập nhật lại ngày kết thúc giao dịch cho mô hình để cho phù hợp với xu thế của thị trường, đồng thời nghiên cứu thêm một số thuật toán của kỹ thuật DRL để cải thiện mô hình trở nên hiệu quả hơn.

## TÀI LIỆU THAM KHẢO


1. Deep Reinforcement Learning for Trading: https://arxiv.org/pdf/1911.10107.pdf
2. Deep reinforcement learning in high frequency trading https://arxiv.org/pdf/2101.07107.pdf
3. Deep Deterministic Policy Gradient (DDPG) : https://spinningup.openai.com/en/latest/algorithms/ddpg.html
4. https://medium.com/mlearning-ai/elegantrl-demo-stock-trading-using-ddpg-part-i-e77d7dc9d208_=
5. https://arxiv.org/pdf/1803.11070.pdf
6. https://arxiv.org/pdf/2011.09607.pdf
7. https://www.reddit.com/r/reinforcementlearning/comments/l3oszi/finrl$_{a}deep_{r}einforcement_{l}earning_{l}ibrary_{f}or/(VideoIntroduce)$
8. Backtest trading Policy : https://github.com/quantopian/pyfolio
9. https://www.google.com/search?q=Dow+jones+30oq=doaqs=chrome.1.69i57j69i59l3j69i60l4.4208j0j7sourceid=chromeie=UTF-8
10. https://gym.openai.com/docs/
11. https://deer.readthedocs.io/en/0.4.1/
12. https://tensorforce.readthedocs.io/en/latest/
13. https://www.tensorflow.org/model$_{o}ptimization$
14. https://pylessons.com/A2C-reinforcement-learning/
15. https://arxiv.org/pdf/1801.01290.pdf
16. https://paperswithcode.com/method/maddpg
17. https://gym.openai.com/
18. https://vietnambiz.vn/ti-le-sharpe-sharpe-ratio-la-gi-cong-thuc-tinh-ti-le-sharpe-20191118095548726.htm