BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

ĐẠI HỌC QUỐC GIA HÀ NỘI

Đề tài

Reinforcement Learning & Knowledge in Learning

(Học tăng cường & Kiến thức trong học tập)

Nhóm 8

Mã lớp : INT3401_20

Giảng viên: TS. Trần Hồng Việt

Hà Nội, 2024

Mục lục

I. Reinforcement Learning

1. Introduction

Phương pháp học củng cố trong trí tuệ nhân tạo, nơi các tác nhân học từ phản hồi về hành vi của mình trong môi trường mà không cần có sự hướng dẫn từ một giáo viên hay tập dữ liệu đã được gán nhãn. Thay vào đó, các tác nhân chỉ nhận được phản hồi sau khi hoàn thành một chuỗi hành động, thường là ở trạng thái kết thúc.

Các điểm quan trọng bao gồm:

Phương pháp học củng cố (Reinforcement Learning Method) : Các tác nhân học từ phản hồi tích cực hoặc tiêu cực sau khi thực hiện các hành động trong một môi trường.

Phần thưởng (Rewards): Phần thưởng là phản hồi từ môi trường, thường được nhận sau khi đạt đến trạng thái kết thúc của một chuỗi hành động.

Học hàm tiện ích và giá trị hành động (Learning Utility and Action Value Functions): Có hai phương pháp chính cho việc học từ phản hồi - học hàm tiện ích trên các trạng thái hoặc học hàm giá trị hành động cho từng hành động cụ thể trong một trạng thái.

Khó khăn và ứng dụng của học củng cố (Challenges and Applications of Reinforcement Learning): Học củng cố là phương pháp phù hợp trong các môi trường phức tạp mà việc đánh giá chính xác từng hành động là khó khăn, và cũng được sử dụng rộng rãi trong nhiều ứng dụng thực tế như huấn luyện robot hoặc các hệ thống tự động chơi trò chơi.

2. Passive learning in a known environment

Hệ thống nhận biết cố gắng học các tiện ích của các trạng thái và chuyển đổi trạng thái. Một mô hình xác suất chuyển đổi giữa các trạng thái được cung cấp.

Trong quá trình huấn luyện, hệ thống nhận biết di chuyển từ trạng thái ban đầu đến trạng thái kết thúc để nhận phần thưởng.

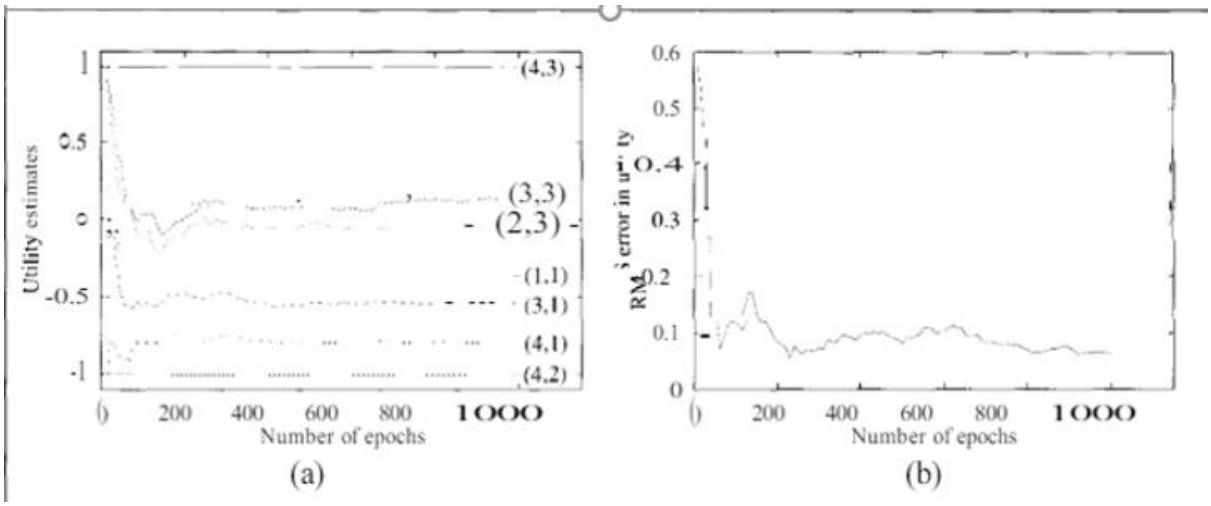
Mục tiêu là học các tiện ích kỳ vọng cho mỗi trạng thái không kết thúc, giả định rằng tiện ích của một chuỗi là tổng các phần thưởng tích lũy trong chuỗi đó.

a) Cập nhật ngây thơ (Naive Updating)

- Phương pháp LMS (Least Mean Squares) được phát triển bởi Widrow và Hoff vào cuối những năm 1950.
- LMS cập nhật ước lượng tiện ích dựa trên phần thưởng quan sát từ mỗi chuỗi huấn luyện.
- Phương pháp này duy trì một trung bình chạy cho mỗi trạng thái để giảm thiểu sai số trung bình bình phương.
- LMS có thể coi là học hàm tiện ích trực tiếp từ các ví dụ, biến bài toán học củng cố thành học có giám sát.
- LMS bỏ qua mối quan hệ phụ thuộc giữa các tiện ích của các trạng thái.

❖ Điều này dẫn đến hội tụ chậm trên các giá trị tiện ích chính xá

Hình minh họa kết quả điển hình trong môi trường 4x3, cho thấy sự hội tụ của ước lượng tiện ích và sự giảm dần của sai số trung bình bình phương so với các giá trị tiện ích chính xác. Hệ thống cần hơn một nghìn chuỗi huấn luyện để tiến gần đến các giá trị chính xác.



Hình 1.Đường cong học tập LMS dành cho thế giới 4x3 được hiển thị trong Hình1. (a) Ước tính tiện ích của các trạng thái theo thời gian, (b) Lỗi RMS so với các giá trị chính xác.

b) Adaptive dynamic programming (Lập trình động thích ứng)

Nhanh hơn khi sử dụng kiến thức cấu trúc môi trường: Học nhanh hơn bằng cách sử dụng kiến thức về xác suất chuyển đổi giữa các trạng thái.

Nhược điểm của LMS: LMS có thể dẫn đến ước lượng tiện ích sai do bỏ qua mối quan hệ giữa các trạng thái

Khắc phục nhược điểm: Sử dụng kiến thức về xác suất chuyển đổi, hệ thống nhận biết có thể giải hệ phương trình tiện ích để tính toán giá trị tiện ích chính xác sau khi quan sát phần thưởng cho tất cả các trạng thái.

Lập trình động điều chỉnh (ADP): Phương pháp học củng cố giải các phương trình tiện ích bằng thuật toán lập trình động. ADP hiệu quả trong việc tận dụng kinh nghiệm, nhưng khó áp dụng cho không gian trạng thái lớn như trong trò chơi backgammon.

c) Temporal difference learning (Học tập khác biệt theo thời gian)

❖ **Xấp xỉ phương trình ràng buộc:** Có thể xấp xỉ các phương trình ràng buộc mà không cần giải chúng cho tất cả các trạng thái.

❖ **Quy tắc cập nhật TD (Temporal Difference):** Sử dụng các chuyển đổi quan sát để điều chỉnh các giá trị trạng thái theo quy tắc: $U(i) \leftarrow U(i) + \alpha(R(i) + U(j) - U(i))$ trong đó α là tham số tốc độ học.

- **Ý tưởng của TD:** Xác định các điều kiện cân bằng khi ước lượng tiện ích là chính xác và sử dụng phương trình cập nhật để dịch chuyển ước lượng đến cân bằng lý tưởng.
- **Cập nhật trạng thái kế cận:** Phương trình cập nhật chỉ liên quan đến trạng thái kế cận thực tế, nhưng giá trị trung bình của U(i) sẽ hội tụ về giá trị chính xác.
- **Thay đổi α :** Giảm dần α khi số lần trạng thái được ghé thăm tăng lên, giúp U(i) hội tụ chính xác (Dayan, 1992).
- **Thuật toán TD-UPDATE:** Sử dụng quy tắc cập nhật TD để điều chỉnh tiện ích.

3. Passive learning in an unknown environment

- **Biểu diễn mạnh mẽ hơn cho hàm tiện ích:** Sử dụng mạng nơ-ron để học trực tiếp từ dữ liệu.
- **Phương pháp LMS và học củng cố:** Bỏ qua mối quan hệ phụ thuộc giữa các tiện ích của các trạng thái.
- **Ràng buộc từ cấu trúc chuyển đổi giữa các trạng thái:** Không xem xét ràng buộc này.
- **Hạn chế của phương pháp LMS:** Dẫn đến hội tụ chậm trên các giá trị tiện ích chính xác.

4. Active learning in an unknown environment

Để điều chỉnh hành động của agent trong mô hình AGENT HỌC CỘNG CỐ, cần thực hiện các thay đổi nhỏ như sau:

- Mô hình môi trường tích hợp xác suất của các chuyển đổi đến các trạng thái khác nhau nếu một hành động cụ thể được thực hiện.
- Các ràng buộc về tiện ích của mỗi trạng thái phải tính đến việc hệ thống nhận biết có lựa chọn các hành động. Sử dụng phương trình:

$$U(i) = R(i) + \max_{a} \sum_{j} M_{ij}^{a} U(j)$$

Cách thay đổi và điều chỉnh thuật toán để áp dụng cho hệ thống nhận biết học cộng cố và học tiện ích hoạt động theo thời gian rời rạc.

- **Hệ thống nhận biết học cộng cố:** Cần có một yếu tố hiệu suất. Thuật toán sẽ cần điều chỉnh để học mô hình môi trường và tiện ích dựa trên hành động được thực hiện. Việc này sẽ yêu cầu một thủ tục để cập nhật mô hình khi hành động được thực hiện và tính toán lại hàm tiện ích.
- **Hệ thống nhận biết học tiện ích theo thời gian rời rạc:** Cần học một mô hình để sử dụng hàm tiện ích để ra quyết định. Tương tự, quy tắc cập nhật vẫn áp dụng nhưng cần đảm bảo rằng các kết quả không thường xuyên được xử lý một cách chính xác trong thuật toán.

5. Exploration

Trong học củng cố tích cực, hệ thống cần quyết định hành động giữa tối ưu hóa phần thưởng hiện tại và học để đạt phần thưởng tương lai:

- Cách tiếp cận "kỳ dị" khám phá ngẫu nhiên, còn "tham lam" tối đa hóa phần thưởng hiện tại.
- Cần cân bằng giữa "kỳ dị" và "tham lam": khám phá nhiều hơn khi không biết về môi trường, khai thác nhiều hơn khi có mô hình gần đúng.
- Gán ước tính tiện ích cao hơn cho các cặp hành động-trạng thái ít được khám phá.
- Phương pháp này tạo niềm tin lạc quan về môi trường, giả định có phần thưởng tuyệt vời ở khắp nơi.

Sử dụng U+(i) cho ước tính lạc quan của tiện ích trạng thái i và N(a, i) cho số lần hành động a được thử trong trạng thái i để tính toán ước tính lạc quan.

$$U^+(i) \leftarrow R(i) + \max_a f\left(\sum_j M_{ij}^a U^+ j),\ \ N(a,i)\right)$$

trong đó f(u,n) được gọi là hàm khám phá.

Nó xác định cách thức mà lòng tham (ưu tiên cho các giá trị u cao) được cân đối với sự tò mò (ưu tiên cho các giá trị n thấp, tức là, các hành động chưa được thử nhiều lần). Hàm f(u,n) nên tăng theo u và giảm theo n.

Rõ ràng, có nhiều hàm khả thi khác nhau phù hợp với những điều kiện này. Một định nghĩa đặc biệt đơn giản là như sau:

$$f(u,n) = \begin{cases} R^+ & \text{if } n < N_e \\ u & \text{otherwise} \end{cases}$$

trong đó:

R+ là ước tính lạc quan về phần thưởng tốt nhất có thể đạt được bất kỳ trạng thái nào

Ne là một tham số cố định.

Điều này sẽ có tác dụng khiến cho hệ thống nhận biết thử mỗi cặp hành động-trạng thái ít nhất Ne lần.

Exploration and bandits

Vấn đề "bandit" mô phỏng việc lựa chọn hành động tối ưu trong một tập hành động có n tùy chọn. Mỗi hành động tương ứng với một phần thưởng có thể khác nhau.

Mục tiêu là tối đa hóa tổng lợi ích kỳ vọng trong suốt cuộc đời. Đối với các vấn đề thực tế, tối ưu hóa có thể trở nên phức tạp và chỉ có thể đạt được kết quả tốt nhất trong các giới hạn kinh nghiệm.

6. Learning an action-value function

Hàm giá trị hành động, gọi là giá trị Q, gán một giá trị kỳ vọng cho việc thực hiện một hành động cụ thể trong một trạng thái cụ thể. Được biểu diễn bằng ký hiệu Q(a, i), nó

liên quan trực tiếp đến giá trị tiện ích tối đa của một trạng thái bằng cách sử dụng phương trình U(i) = max_a Q(a, i).

Giá trị Q đóng vai trò quan trọng trong học tăng cường vì chúng cho phép đưa ra quyết định mà không cần một mô hình, và chúng có thể được học trực tiếp từ phản hồi phần thưởng.

Giống như với các tiện ích, chúng ta có thể viết một phương trình ràng buộc phải thỏa mãn tại điểm cân bằng khi các giá trị Q là chính xác:

$$Q(a,i) = R(i) + \sum_j M_{ij}^a \max_{a'} Q(a',j)$$

Phương pháp Q-learning sử dụng phương trình cập nhật chênh lệch thời gian để tính toán giá trị Q mà không cần một mô hình. Phương trình cập nhật cho Q-learning theo phương pháp TD là

$$Q(a,i) \leftarrow Q(a,i) + \alpha\,(R(i) + \max_{a'} Q(a',j) - Q(a,i))$$

Tác nhân Q-learning sử dụng phương pháp chênh lệch thời gian để cập nhật giá trị Q mà không cần một mô hình, tạo ra một hệ thống linh hoạt và không đòi hỏi kiến thức trước.

7. Generalization in reinforcement learning

Phương pháp học cảm ứng thường sử dụng biểu diễn ngầm cho các hàm tiện ích và giá trị hành động, thay vì biểu diễn dưới dạng bảng, để giảm chi phí tính toán và tăng khả năng tổng quát hóa. Hàm tiện ích có thể được biểu diễn như một hàm tuyến tính của các đặc trưng của môi trường, như trong phương trình

U(i) = w1f1(i) + w2f2(i) + ... + wnfn(i).

Điều này giúp giảm kích thước của không gian giả thuyết và tăng khả năng tổng quát hóa đầu vào.

Phương trình cập nhật của thuật toán TD trong học cảm ứng có thể được biểu diễn như sau: $w \leftarrow w + \alpha\,[\,r + U_w(j) - U_w(i)\,]\;\nabla_w U_w(i)$

Đây là một quy tắc cập nhật gradient descent trong không gian trọng số, nhằm giảm thiểu sai lệch cục bộ trong ước tính tiện ích U.

8. Genetic algorithms and evolutionary programming

Cuốn sách The Origin of Species của Darwin đã định nghĩa một cách tiến hóa tự nhiên thông qua nguyên tắc lựa chọn tự nhiên, và nguyên tắc này cũng được áp dụng trong các hệ thống nhân tạo như thuật toán GENETIC-ALGORITHM

Thuật toán này bắt đầu với một tập hợp các cá thể và sử dụng các toán tử lựa chọn và sinh sản để tiến hóa các cá thể thành công dựa trên một hàm thích nghi

Trong thuật toán GENETIC-ALGORITHM, chiến lược lựa chọn thường là ngẫu nhiên, với xác suất lựa chọn tỉ lệ thuận với thích nghi của mỗi cá thể.

Giải thuật di truyền là một công cụ linh hoạt và đơn giản để giải quyết các vấn đề, nhưng hiệu suất của nó có thể thay đổi đáng kể trên các vấn đề cụ thể

9. Tóm tắt tổng quan

Thiết kế tác nhân có thể dựa trên mô hình hoặc không cần mô hình, sử dụng hàm tiện ích hoặc hàm giá trị hành động.

Tiện ích của một trạng thái là tổng kỳ vọng của các phần thưởng từ bây giờ đến khi kết thúc.

Có ba phương pháp chính để học tiện ích: LMS, ADP và TD.

Phương pháp TD làm đơn giản hóa việc học Q mà không cần mô hình, nhưng có thể hạn chế trong các môi trường phức tạp.

Trong không gian trạng thái lớn, cần sử dụng biểu diễn hàm mặc định và tín hiệu TD có thể điều chỉnh trọng số trong mạng nơron.

II. Knowledge in learning

1. Introduction

Đặc trưng logic và học cảm ứng: Đặc trưng logic của vấn đề học tập cho phép chỉ định thông tin về hàm cần học.

Mục tiêu của học cảm ứng: Tìm giả thuyết giải thích phân loại dựa trên các mô tả, được biểu diễn logic bằng công thức "Hypothesis A Descriptions |= Classifications"

a) Một số ví dụ đơn giản

Nướng Thằn Lằn: nướng thằn lằn bằng đầu cây, dẫn đến kết luận rằng bất kỳ vật thể nào cũng có thể dùng để nướng thực phẩm mềm nhỏ.

Du Khách Ở Brazil: Một du khách kết luận rằng người Brazil nói tiếng Bồ Đào Nha dựa trên rằng mọi người trong một quốc gia thường nói cùng một ngôn ngữ.

b) Một số sơ đồ chung

Nướng Thằn Lằn: Học dựa trên giải thích (EBL) biến nguyên tắc đầu tiên thành kiến thức cụ thể.

Du Khách Ở Brazil: Học dựa trên sự liên quan (RBL) cho phép du khách kết luận về ngôn ngữ nói chung của người Brazil dựa trên một trường hợp cụ thể.

2. Explanation-based learn

Học dựa trên giải thích (EBL) là một phương pháp học tập dựa trên việc trích xuất các quy tắc tổng quát từ các quan sát cá nhân.

Ví dụ, khi phân biệt một biểu thức như X^2 theo X, ta thu được 2X. EBL cho phép tổng quát hóa quy trình này thành các quy tắc tổng quát, như "đạo hàm của u^2 theo u là 2u".

Khi một nguyên tắc được hiểu, nó có thể được tổng quát hóa và tái sử dụng trong các tình huống khác, giúp giải quyết các vấn đề phức tạp hơn.

3. Extracting general rules from examples

Học dựa trên giải thích (EBL) xây dựng quy tắc tổng quát từ các chứng minh cụ thể, loại bỏ điều kiện không cần thiết để tạo ra quy tắc hiệu quả hơn.

Quy trình cơ bản của EBL hoạt động như sau:

- Với một ví dụ cụ thể, xây dựng một chứng minh rằng mục tiêu áp dụng cho ví dụ đó bằng cách sử dụng kiến thức nền tảng có sẵn.
- Đồng thời, xây dựng một cây chứng minh tổng quát cho mục tiêu biến đổi bằng cách sử dụng các bước suy luận giống như trong chứng minh ban đầu.
- Xây dựng một quy tắc mới mà phía bên trái của nó bao gồm các lá của cây chứng minh, và phía bên phải là mục tiêu biến đổi.
- Loại bỏ bất kỳ điều kiện nào đúng bất kể giá trị của các biến trong mục tiêu.

Improving efficiency

- **Cải thiện hiệu suất của EBL:**
  - ✓ Cân nhắc giữa tính khả thi và tính tổng quát của các quy tắc trích xuất.
  - ✓ Quy tắc phải hữu ích trong việc giải quyết các trường hợp cụ thể, tránh ngõ cụt và rút ngắn chứng minh.
  - ✓ Thêm quá nhiều quy tắc có thể làm chậm quá trình suy luận, cần đánh giá hiệu suất và tính tổng quát.
- **Đảm bảo tính khả thi:**
  - ✓ Đảm bảo mục tiêu phụ trong quy tắc khả thi.
  - ✓ Mục tiêu phụ cụ thể dễ giải quyết hơn nhưng bao quát ít trường hợp hơn.
  - ✓ Đánh giá tính khả thi phải xem xét độ phức tạp, chi phí giải quyết và ảnh hưởng đến cơ sở tri thức tổng thể.
- **Phân tích thực nghiệm:**
  - ✓ EBL tạo cơ sở tri thức hiệu quả hơn từ các ví dụ trong quá khứ.
  - ✓ Hiệu suất cải tiến giả định phân phối của các ví dụ gần với các vấn đề tương lai.
  - ✓ Cải tiến hiệu suất đáng kể cho các vấn đề tương tự trong tương lai.

4. Learning using relevance informat

Learning Using Relevance Information (LURI) là một phương pháp học máy sử dụng thông tin về mức độ liên quan của các đặc điểm (features) để cải thiện quá trình học.

a) Determining the hypothesis space

Xác định chức năng giữa các thuộc tính giúp hạn chế không gian các giả thuyết, làm cho việc học dễ dàng hơn. Sử dụng kết quả cơ bản của lý thuyết học máy, ta có thể định lượng lợi ích của việc này.

b) Learning and using relevance information

Thuật toán học được trình bày trong phần này tìm cách tìm ra xác định đơn giản nhất phù hợp với các quan sát.

5. Inductive logic programming

Lập trình logic trong trí tuệ nhân tạo (ILP) kết hợp phương pháp thuận dụng và biểu diễn bậc nhất, tập trung vào biểu diễn lý thuyết dưới dạng chương trình logic.

a) Inverse resolution

Ngược định lý trong học dựa trên logic cho phép tìm ra giả thuyết phù hợp bằng cách chạy ngược lại quá trình giải quyết.

- **Generating inverse proofs**

Là phương pháp tách mệnh đề hợp thành các mệnh đề hoặc ngược lại, là một quá trình tìm kiếm đòi hỏi lựa chọn mỗi bước, thường được cải thiện bằng cách áp dụng các hạn chế và chiến lược giải quyết.

- **Discovering new predicates and new knowledge**

Dựa trên logic có thể tạo ra các giả thuyết từ các ví dụ bằng cách đảo ngược các chiến lược giải quyết, thậm chí cung cấp khả năng phát minh các predicate mới để giải thích dữ liệu.

b) Top-down learning methods

Phương pháp học từ trên xuống trong ILP tập trung vào việc tạo ra các quy tắc tổng quát từ một quy tắc khởi đầu và điều chỉnh chúng để phù hợp với dữ liệu.

FOIL là một ví dụ điển hình, xây dựng các mệnh đề từng bước một và sử dụng một phép chọn lọc để tạo ra các quy tắc phân loại đúng cho các ví dụ.

6. Tóm tắt tổng quan

Chương này khám phá cách sử dụng kiến thức trước đó để giúp đại lý học từ các trải nghiệm mới.

Điều này bao gồm việc tích lũy kiến thức, loại bỏ các giả thuyết không nhất quán, hiểu vai trò của logic trong kiến thức trước đó, và sử dụng các phương pháp như EBL, RBL, KBIL và ILP để chuyển đổi kiến thức nguyên lý thành kiến thức chuyên môn hiệu quả.

Các phương pháp ILP cụ thể tạo ra các loại mới và hứa hẹn trong việc biểu diễn các lý thuyết mới trong nhiều lĩnh vực khác nhau.