ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

Lê Kim Thư

PHÁT TRIỂN MÔ HÌNH THAY THẾ AXIT AMIN CHO DỮ LIỆU HỆ GEN

Ngành đào tạo: Khoa học máy tính

Mã số: 9840101

TÓM TẮT LUẬN ÁN TIẾN SĨ KHOA HỌC MÁY TÍNH

Hà Nội – 2024

Công trình được hoàn thành tại: Trường Đại học Công nghệ, Đại học Quốc gia Hà Nội

Người hướng dẫn khoa học: PGS.TS. Lê Sỹ Vinh

Phản biện: .....
.....
Phản biện: .....
.....
Phản biện: .....
.....

Luận án sẽ được bảo vệ trước Hội đồng cấp Đại học Quốc gia chấm luận án tiến sĩ họp tại .....
vào hồi giờ ngày tháng năm

Có thể tìm hiểu luận án tại:

- Thư viện Quốc Gia Việt Nam
- Trung tâm Thông tin - Thư viện, Đại học Quốc gia Hà Nội
-

Mở đầu

Tính cấp thiết của luận án

Luận án nghiên cứu tối ưu mô hình tiến hóa sử dụng trong bài toán xây dựng cây phân loài bằng phương pháp cực đại khả năng (**ML**).

Cây phân loài là một đồ thị dạng cây thể hiện mối quan hệ giữa các loài được nghiên cứu. Xây dựng cây phân loài là bài toán trung tâm của tin sinh học. Hiện nay, các bộ dữ liệu trình tự có kích thước lớn ngày càng trở nên phổ biến, đây là kết quả của sự phát triển công nghệ sinh học hiện đại. Việc sử dụng các mô hình chung cho toàn thể sinh vật hoặc chỉ sử dụng một mô hình cho tất cả các vị trí trên trình tự có thể không xây dựng được cây tiến hóa tốt. Từ đó nảy sinh nhu cầu về mô hình thay thế mới để biểu diễn quá trình tiến hóa cho một nhóm sinh vật cụ thể và mô hình biểu diễn sự không đồng nhất trong quá trình tiến hóa tại các vị trí khác nhau trên trình tự.

Mục tiêu của luận án

Mục tiêu nghiên cứu của luận án là trả lời và tìm giải pháp cải tiến cho các câu hỏi:

1. Biểu diễn quá trình tiến hóa trên dữ liệu lớn như thế nào để nâng cao tính đúng đắn của cây phân loài?
2. Các mô hình tiến hóa chung có phù hợp để sử dụng cho bộ dữ liệu của một nhóm sinh vật cụ thể không?
3. Chỉ sử dụng một mô hình tiến hóa cho toàn bộ dữ liệu có cho kết quả tốt không?
4. Làm thế nào để giảm thời gian thực hiện cho các bộ dữ liệu có kích thước rất lớn?

Các đóng góp của luận án

Luận án đã đạt được một số kết quả và đóng góp như sau :

1. Luận án đề xuất mô hình thay thế axit amin cho vi rút trong chi Flavivirus và thực hiện phân tích, đánh giá tính hiệu quả của mô hình khi xây dựng cây phân loài so với các mô hình sẵn có.
2. Đề xuất một số thuật toán để phân hoạch tập vị trí cho một bộ dữ liệu cho trước. Mỗi tập con của phân hoạch thể hiện một nhóm vị trí có quá trình tiến hóa tương đồng trên hệ gen, các tập con khác nhau có thể được gán một mô hình tiến hóa khác nhau do vậy quá trình tiến hóa được biểu diễn gần với thực tế hơn.
3. Đề xuất một thuật toán ước lượng nhanh tốc độ biến đổi tương đối giữa các vị trí trong sắp hàng.

Các kết quả liên quan tới luận án đã được công bố trong 2 bài báo tạp chí SCIE, 1 bài báo tạp chí đại học quốc gia và 5 bài hội nghị quốc tế.

Bố cục của luận án

Ngoài phần Mở đầu, Kết luận và tài liệu tham khảo, luận án gồm 4 chương, được tổ chức như sau:

Chương 1 giới thiệu các khái niệm cơ bản và một số kiến thức nền tảng quan trọng bao gồm bài toán xây dựng cây phân loài và các vấn đề liên quan. Chương 1 cũng giới thiệu các phương pháp thường dùng để đánh giá kết quả nghiên cứu.

Chương 2 trình bày quá trình ước lượng mô hình thay thế axit amin FLAVI cho chi Flavivirus và các thực nghiệm đã thực hiện để so sánh mô hình FLAVI với các mô hình hiện có.

Chương 3 đề xuất thuật toán phân hoạch sắp hàng mPartition. Thuật toán phân hoạch tập vị trí của sắp hàng dựa trên thông tin (i) tốc độ tiến hóa (ii) mô hình tiến hóa phù hợp nhất tại mỗi vị trí. Cuối chương trình bày thiết lập thực nghiệm và kết quả.

Chương 4 đề xuất (i) thuật toán ước lượng nhanh tốc độ biến đổi fastTIGER và (ii) đề xuất thuật toán gPartition phân hoạch nhanh sắp hàng cỡ hệ gen. Hai thuật toán đều có thể thực hiện trên các sắp hàng có độ dài lên đến hàng triệu vị trí trong thời gian chấp nhận được. Cuối chương trình bày thiết lập thực nghiệm và kết quả.

Chương 1: Cơ sở lý thuyết

1.1 Các khái niệm cơ bản

1.1.1 Thông tin di truyền

1.1.2 Các biến đổi di truyền

1.1.3 Mô hình thay thế nucleotit/axit amin

1.1.4 Sắp hàng đa trình tự tương đồng

1.1.5 Cây phân loài

1.1.6 Xây dựng cây phân loài bằng phương pháp cực đại khả năng

1.2 Bài toán ước lượng mô hình thay thế axit amin

1.2.1 Bài toán

Đầu vào: N sắp hàng axit amin ký hiệu là $\mathbf{D} = \{D_1, \dots, D_N\}$.

Bài toán: Ước lượng mô hình thay thế axit amin mô tả xác suất biến đổi giữa các axit amin trong quá trình tiến hóa ứng với dữ liệu trong **D**

Đầu ra: Mô hình thay thế axit amin Q biểu diễn quá trình biến đổi axit amin trên các trình tự của bộ dữ liệu **D**.

Yêu cầu: Phương pháp ước lượng cần thu được mô hình có độ chính xác cao và thực hiện trong thời gian chấp nhận được.

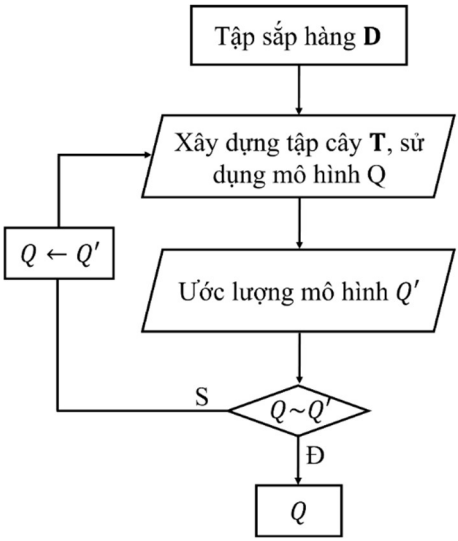
1.2.2 Các phương pháp ước lượng mô hình thay thế axit amin

Các phương pháp ước lượng mô hình thay thế axit amin có thể chia làm hai loại chính là phương pháp đếm và phương pháp cực đại khả

năng. Phương pháp LA sử dụng là phương pháp cực đại khả năng được thực hiện như sau:

Với đầu vào $\mathbf{D} = (D_1, \dots, D_N)$. Gọi $\mathbf{T} = (T_1, \dots T_N)$ là tập các cây tương ứng với các sắp hàng trong **D**. Phương pháp ước lượng cực đại khả năng xác định tập cây **T**, mô hình Q để cực đại hóa giá trị khả năng (likelihood) $L(Q, \mathbf{T}|\mathbf{D})$

Tuy nhiên việc tối ưu các tham số cây và mô hình cùng lúc là rất khó khăn nên thay vì tối ưu cùng lúc cả Q và **T**, ta sẽ tìm tập cây gần tối ưu **T** trước, sau đó cố định **T** và tìm mô hình Q. Quá trình thực hiện như trong Hình 1.3



Hình 1.3 Sơ đồ quá trình ước lượng mô hình thay thế bằng phương pháp ML

1.3 Tính không đồng nhất của quá trình biến đổi tại các vị trí khác nhau

Trong thực tế, quá trình tiến hóa hay chính là quá trình thay đổi trạng thái tại các vị trí khác nhau có thể khác nhau, hiện tượng này gọi là tính không đồng nhất của quá trình biến đổi tại các vị trí khác nhau. Nhiều phương pháp khác nhau được đề xuất để biểu diễn tính không đồng nhất này. Yêu cầu của các phương pháp là biểu diễn được quá trình tiến hóa không đồng nhất giữa các vị trí mà vẫn đảm bảo khối lượng tính toán hợp lý.

1.3.1 Mô hình tốc độ biến đổi

Mô hình tốc độ biến đổi ký hiệu là V sử dụng một phân phối gamma rời rạc kỳ vọng 1.0, phương sai $1/\alpha$ để tính tham số tốc độ, giá trị tham số được nhân vào xác suất chuyển trạng thái tức thì trong mô hình Q. Hàm phân phối thường sử dụng 4 nhóm tốc độ với xác suất đều nhau. Giá trị trung bình hoặc trung vị của mỗi nhóm được sử dụng làm tham số tốc độ cho tất cả các tốc độ trong nhóm đó.

1.3.2 Mô hình đa ma trận

1.3.3 Mô hình lược đồ phân vùng

Với sắp hàng D, một "*lược đồ phân vùng*" ('partitioning scheme') của D, ký hiệu là $\mathbf{PS} = \{S_1, \dots, S_k\}$, là một phân hoạch trên tập vị trí của sắp hàng D. Mỗi tập con S_i (có l_i vị trí) của phân hoạch được gọi là một phân vùng (partition) vị trí.

Ký hiệu mô hình thay thế phù hợp nhất và mô hình tốc độ biến đổi của phân vùng S_i là Q_i và V_i; tập hợp mô hình thay thế và mô hình tốc độ biến đổi cho cả lược đồ là $\mathbf{Q}$ và $\mathbf{V}$. Khi đó, giá trị khả năng của sắp hàng được xác định bằng:

$$L(D|T, \mathbf{Q}, \mathbf{V}) = \prod\prod\prod L(s_{ij}|T, Q_i, V_i) = \prod\prod\prod P(s_{ij}|T, Q_i, V_i)$$

1.3.4 Một số thuật toán phân hoạch sắp hàng

Mục tiêu của các thuật toán phân hoạch sắp hàng là tìm một lược đồ phân vùng thể hiện tốt nhất sự không đồng nhất trong quá trình biến đổi giữa các nucleotit/axit amin tại các vị trí khác nhau trên sắp hàng. Mô hình lược đồ phân vùng được đánh giá thông qua cây phân loài cực đại khả năng cho sắp hàng D xây dựng được khi sử dụng mô hình này.

Phần này trình bày hai thuật toán tự động phân hoạch sắp hàng *k*-means lặp và RatePartition là hai thuật toán phân hoạch sắp hàng dựa trên tốc độ biến đổi tại mỗi vị trí. Thuật toán *k*-means lặp luôn tạo ra phân vùng bất biến – là phân vùng chứa tất cả và chỉ chứa các vị trí không có biến đổi trong dữ liệu, do vậy thuật toán không được sử dụng nữa. Thuật toán RatePartition sử dụng một công thức đơn giản để chia đoạn giá trị tốc độ do vậy đã bổ sung thêm một số vị trí biến đổi chậm vào phân vùng bất biến nói trên.

1.4 Tốc độ tiến hóa tại mỗi vị trí trên sắp hàng

Một cách tổng quát, tốc độ tiến hóa là tốc độ mà các loài sinh vật thay đổi và thích nghi với môi trường thể hiện qua quá trình tiến hóa. Với phạm vi dữ liệu trong một sắp hàng, tốc độ tiến hóa là tương đối giữa các vị trí; một số vị trí có thể phát triển với tốc độ nhanh hơn trong khi vị trí khác phát triển chậm hơn.

Phần này trình bày thuật toán TIGER ước lượng tốc độ tiến hóa tại mỗi vị trí mà không sử dụng một cây đầu vào. Cách tính này không sử dụng cây và được đánh giá tốt.

1.5 Các phương pháp đánh giá mô hình

1.5.1 So sánh giá trị khả năng của cây phân loài xây dựng bằng phương pháp cực đại khả năng

1.5.2 Các độ đo AIC và BIC

1.5.3 So sánh cấu trúc cây

1.6 Kết luận

Trong quá trình xác định mối quan hệ giữa các loài thông qua việc xây dựng cây phân loài bằng phương pháp cực đại khả năng, việc lựa chọn mô hình thay thế phù hợp nhất có ảnh hưởng lớn đến cây phân loài xây dựng được. Các mô hình thay thế mới cần được nghiên cứu và đề xuất nhằm nâng cao khả năng biểu diễn dữ liệu so với các mô hình đã có.

Với những bộ dữ liệu lớn, vấn đề tiến hóa không đồng nhất tại các vị trí khác nhau trên trình tự cần được tính đến khi xây dựng cây. Lược đồ phân vùng là phương pháp hiệu quả về mặt tính toán để nâng cao tính đúng đắn của cây phân loài xây dựng được. Các thuật toán tự động phân hoạch sắp hàng hiện nay chỉ sử dụng tốc độ tiến hóa, không phản ánh hết quá trình tiến hóa phức tạp do vậy cần được nghiên cứu và cải tiến thêm.

Chương 2: Mô hình thay thế axit amin FLAVI cho Flavivirus

2.1 Giới thiệu

2.2 Phương pháp

So với phương pháp đã trình bày trong 1.2.2, LA sử dụng thêm mô hình tốc độ biến đổi **V** để biểu diễn tính không đồng nhất trong dữ liệu

và áp dụng phương pháp ước lượng nhanh FastMG để tăng tốc độ thực hiện mà vẫn đảm bảo tính đúng đắn

Algorithm Phương pháp chia sắp hàng dựa trên cấu trúc cây

input: Sắp hàng $D = \{d_1; d_2; ...; d_n\}$ và số nguyên dương k
output: Tập các sắp hàng con $\mathbf{L} = \{D_1; D_2; ...; D_m\}$; D_i có từ $k \to 2k$ trình tự.
procedure TreeBasedSplit(D, k)
    Khởi tạo $\mathbf{G} = \{G_i = d_i\}$; $\mathbf{L} = \emptyset$
    while Còn > 1 nhóm trong $\mathbf{G}$ **do**
        for all $i \neq j$ **do**
            Tính khoảng cách giữa G_i và G_j sử dụng thuật toán BIONJ [18]
        end for
        Tìm hai nhóm có khoảng cách nhỏ nhất, giả sử là G_1 và G_2, số trình tự trong mỗi nhóm là m_1 và m_2
        if $m_1 + m_2 \leq 2k$ **then**
            Gộp G_2 vào G_1
        else
            if $m_1 > k$ **then**
                $\mathbf{G} = \mathbf{G} \backslash G_1$
                $\mathbf{L} = \mathbf{L} \cup G_1$
            else
                $\mathbf{G} = \mathbf{G} \backslash G_2$
                $\mathbf{L} = \mathbf{L} \cup G_2$
            end if
        end if
    end while
    if Số trình tự trong nhóm còn lại > 2 **then**
        Coi nhóm còn lại là một sắp hàng con và thêm vào $\mathbf{L}$
    else
        Nhập nhóm còn lại vào nhóm gần nhất trong $\mathbf{L}$
    end if
end procedure

Thuật toán 2. 1 Phương pháp chia sắp hàng dựa trên cấu trúc cây BIONJ

của mô hình, cụ thể là chia nhỏ sắp hàng ban đầu có số lượng trình tự lớn thành các sắp hàng con ít trình tự hơn, việc chia nhỏ dựa trên cấu trúc cây BIONJ. Thuật toán chia nhỏ sắp hàng thực hiện như

trong Thuật toán 2.1. Bên cạnh đó, LA cũng sử dụng tập mô hình khởi tạo để khoanh vùng cho tác vụ chọn mô hình phù hợp nhất.

Thuật toán ước lượng thực hiện như trong Thuật toán 2.2.

2.3 Thực nghiệm

2.3.1 Dữ liệu

2.3.2 Tham số cài đặt

2.3.3 Thực nghiệm

2.4 Kết quả

2.4.1 Tính bền vững của mô hình

2.4.2 Phân tích và đánh giá mô hình

2.4.3 So sánh hiệu quả của FLAVI

2.5 Kết luận

FLAVI khác biệt với các mô hình đã có và sử dụng mô hình FLAVI giúp xây dựng cây ML tốt hơn cho các vi rút trong chi Flavivirus. Do vậy, để tiết kiệm thời gian, các nhà khoa học có thể sử dụng FLAVI là mô hình mặc định khi phân tích trình tự protein của các vi rút trong chi Flavivirus.

Chương 3: Phương pháp phân hoạch sắp hàng sử dụng mô hình tiến hóa

3.1 Giới thiệu

3.2 Phương pháp

Thuật toán 2. 2 Ước lượng nhanh mô hình thay thế axit amin cho bộ dữ liệu **D**

Đầu vào của bài toán là sắp hàng đa trình tự tương đồng $D = \{d_1, \dots, d_l\}$ độ dài l, với d_i là một cột của ma trận dữ liệu D.

Đầu ra là lược đồ phân vùng $\mathbf{PS} = \{P_1, \dots, P_k\}$ là một phân hoạch của tập các vị trí từ 1 đến l.

Thuật toán mPartition thực hiện tìm lược đồ phân vùng sao cho cây cực đại khả năng xây dựng cho sắp hàng D sử dụng lược đồ phân vùng tìm được có điểm BIC nhỏ nhất.

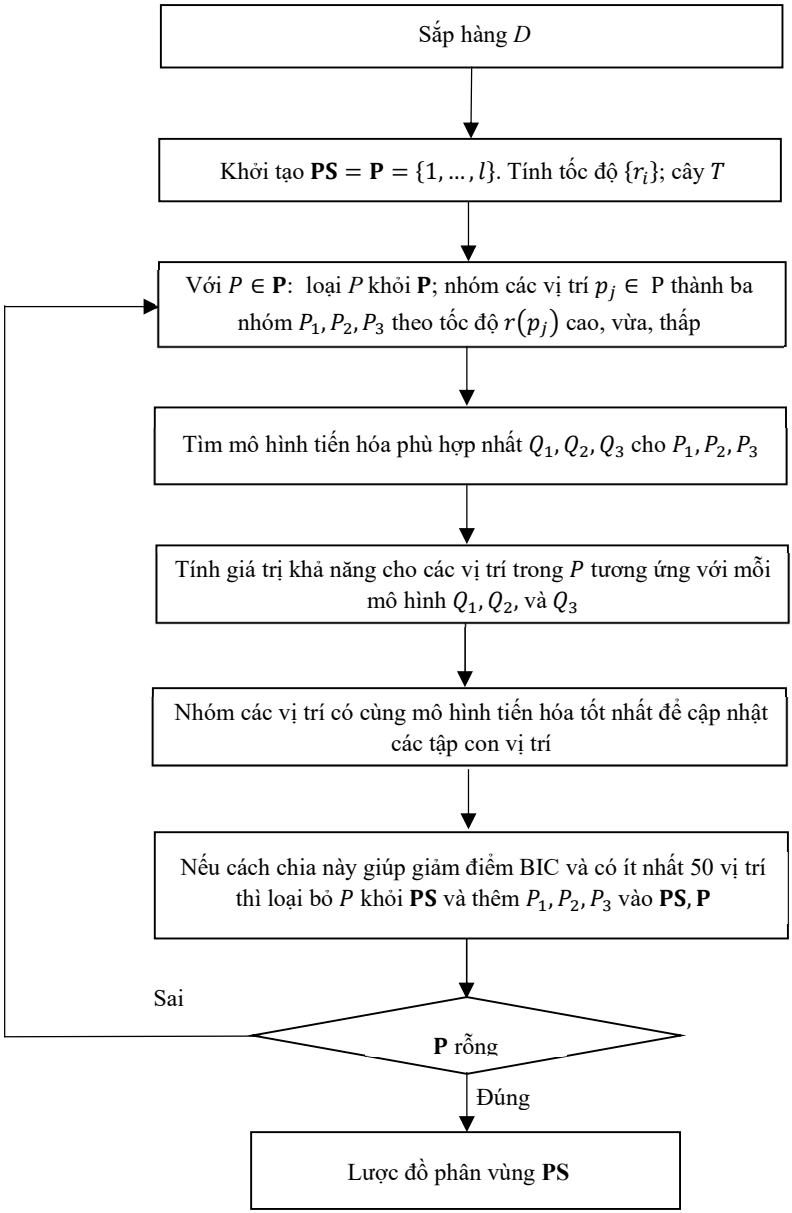
mPartition bắt đầu bằng lược đồ khởi tạo chỉ có một phân vùng, chứa tất cả các vị trí của sắp hàng. Với lược đồ hiện tại, thuật toán chia nhỏ các phân vùng vị trí trong đó thành các phân vùng nhỏ hơn để thu được lược đồ mới tốt hơn (điểm BIC của cây cực đại khả năng giảm).Quy trình trên lặp lại đến khi không thể chia nhỏ các phân vùng để làm giảm điểm BIC.

Để chia nhỏ một phân vùng, trước hết mPartition chia các vị trí trong phân vùng đang xét thành ba tập con: nhóm tiến hóa chậm gồm các vị trí có giá trị tốc độ lớn, nhóm tiến hóa nhanh có giá trị tốc độ nhỏ, còn lại là nhóm tiến hóa trung bình (tốc độ ước lượng bởi thuật toán TIGER). Sau đó tìm mô hình phù hợp nhất cho các nhóm và điều chỉnh lại tập vị trí trong mỗi phân vùng theo mô hình phù hợp nhất (vị trí có biến đổi) và hàm ánh xạ giá trị khả năng của các cặp vị trí – mô hình tiến hóa [75] (vị trí bất biến) .

Thuật toán mPartition gồm bốn bước và được thực hiện như trong Hình 3.1.

3.3 Thực nghiệm

Các thực nghiệm được thực hiện được thực hiện để so sánh mPartition và RatePartition trên 3 loại dữ liệu bao gồm dữ liệu



Hình 3. 1 Quy trình phân vùng vị trí bằng phương pháp mPartition

DNA mô phỏng, dữ liệu DNA thực và dữ liệu Protein thực. Các bộ dữ liệu được lấy từ các nghiên cứu đã được công bố

3.3.1 Dữ liệu

3.3.2 Thực nghiệm

3.4 Kết quả

3.4.1 Dữ liệu DNA mô phỏng

3.4.2 Dữ liệu DNA thực

3.4.3 Dữ liệu protein thực

3.5 Kết luận

Thuật toán áp dụng được cho nhiều loại dữ liệu, kết quả của thuật toán là lược đồ phân vùng được sử dụng trong quá trình xây dựng cây phân loài nhằm tăng tính chính xác của cây phân loài xây dựng được.

Chương 4: Phương pháp phân hoạch sắp hàng cho dữ liệu hệ gen

4.1 Mở đầu

4.2 Phương pháp

4.2.1 Thuật toán ước lượng nhanh tốc độ tiến hóa

4.2.2 gPartition

4.3 Thực nghiệm

4.3.1 Dữ liệu

4.3.2 Tham số cài đặt

4.3.3 Thực nghiệm

4.4 Kết quả

Các kết quả chính thu được qua thực nghiệm là:

- Tốc độ do thuật toán fastTIGER tính toán có hệ số tương quan vừa đến cao với tốc độ tính bởi TIGER.

- Khảo sát trên hai thuật toán mPartition và RatePartition khi dùng tốc độ TIGER và fastTIGER cho kết quả tương đồng.
- Thời gian tính toán của fastTIGER nhanh gấp nhiều lần so với TIGER, đặc biệt là trên các sắp hàng lớn.
- Thuật toán gPartition tạo lược đồ phân vùng tốt hơn so với RatePartition chỉ dùng tốc độ tiến hóa nhưng không tốt bằng thuật toán mPartition trong một số trường hợp (tập trung ở các sắp hàng rất lớn).
- gPartition có tốc độ chạy nhanh, có thể xử lý những sắp hàng lớn với hàng triệu nucleotit trong thời gian cho phép (các sắp hàng trong phạm vi thực nghiệm đều được phân hoạch trong không quá 24 giờ). Những bộ này không thể xử lý bằng mPartition.

4.5 Kết luận

Chương 4 đã trình bày hai thuật toán: thuật toán ước lượng nhanh tốc độ tiến hóa fastTIGER và thuật toán gPartition để phân hoạch những sắp hàng DNA lớn và rất lớn.

Các phân tích thực nghiệm cho thấy fastTIGER có thời gian thực hiện ngắn hơn nhiều so với TIGER và tốc độ tính bằng thuật toán này có thể thực hiện cho các thuật toán phân hoạch sắp hàng.

Thuật toán gPartition có thể phân hoạch sắp hàng DNA khổng lồ trong thời gian cho phép. Lược đồ tạo bởi gPartition tốt hơn so với RatePartition, tuy nhiên trong nhiều trường hợp không tốt bằng lược đồ kết quả của mPartition.

Kết luận

Xây dựng cây phân loài sát với thực tế rất quan trọng trong các bài toán phân tích dữ liệu sinh học phân tử và là một trong những vấn đề chính của tin sinh học. Sử dụng mô hình tiến hóa không phù hợp với

bản chất của dữ liệu có thể ảnh hưởng đáng kể đến tính đúng đắn của cây phân loài cực đại khả năng xây dưng được. Do đó, các mô hình tiến hóa mới và cách thức biểu diễn mới được nghiên cứu, đề xuất để mô tả tốt nhất quá trình tiến hóa trong một bộ dữ liệu.

Luận án tập trung vào việc nghiên cứu và đưa ra các đề xuất làm tăng tính đúng đắn khi biểu diễn quá trình tiến hóa trong một sắp hàng trên hai khía cạnh: một là đề xuất mô hình đơn biểu diễn tốt hơn quá trình tiến hóa của một nhóm dữ liệu (chi Flavivirus), hai là đề xuất thuật toán phân hoạch để biểu diễn quá trình tiến hóa không đồng nhất của dữ liệu nói chung.

Luận án đã đề xuất mô hình thay thế FLAVI dành riêng cho các loài vi rút trong chi một chi chứa nhiều loại vi rút gây dịch bệnh nghiêm trọng trong khu vực. Thực nghiệm cho thấy mô hình FLAVI giúp xây dựng cây cực đại khả năng tốt hơn đáng kể so với các mô hình hiện có.

Luận án cũng đề xuất hai phương pháp mPartition và gPartition để phân hoạch sắp hàng kích thước lớn và rất lớn. Hai phương pháp đều sử dụng kết hợp tốc độ tiến hóa và mô hình tiến hóa tại từng vị trí để phân hoạch tập vị trí, giúp tạo lược đồ phân vùng tốt hơn so với thuật toán chỉ sử dụng tốc độ tiến hóa. Tuy nhiên các thuật toán được đề xuất vẫn còn một số nhược điểm chưa giải quyết được: thuật toán mPartition có thể chạy được với mọi loại dữ liệu nhưng chưa áp dụng được cho dữ liệu hệ gen, ngược lại, gPartition áp dụng được cho dữ liệu hệ gen nhưng lại chỉ dùng được cho dữ liệu DNA.

Bên cạnh đó, luận án đề xuất phương pháp ước lượng nhanh tốc độ tiến hóa fastTIGER. Phương pháp ước lượng tốc độ mới có độ phức tạp nhỏ hơn và có thể tính ra kết quả trong thời gian rất ngắn. Ngoài việc ứng dụng trong các thuật toán phân hoạch sắp hàng, phương pháp

ước lượng nhanh tốc độ tiến hóa fastTIGER cũng có những ứng dụng riêng; ví dụ: kiểm tra và xóa những vị trí có mẫu đặc biệt để giảm tính dị biệt của các vị trí trong sắp hàng.

Danh mục các công trình khoa học của tác giả liên quan đến luận án

[CT 1] Le Kim Thu, Cuong Dang Cao, and Vinh Le Sy . 2018. "Building a specific amino acid substitution model for dengue viruses." 2018 10th International Conference on Knowledge and Systems Engineering (KSE) 242-246.

[CT 2] Le Kim Thu, and Vinh Le Sy. 2020. "A protein alignment partitioning method for protein phylogenetic inference." 2020 RIVF International Conference on Computing and Communication Technologies (RIVF) 1-5.

[CT 3] Le Kim Thu, Vinh Le Sy, Dong Do Duc, Thang Bui Ngoc, and Phuong Thao Nguyen Thi. 2020. "iK-means: an improvement of the iterative k-means partitioning algorithm." 2020 12th International Conference on Knowledge and Systems Engineering (KSE) 300-305.

[CT 4] Le Kim Thu, and Vinh Le Sy. 2020. "FLAVI: An Amino Acid Substitution Model for Flaviviruses." Journal of molecular evolution 88 (5): 445-452.

[CT 5] Le Kim Thu, and Vinh Le Sy. 2020. "mPartition: A Model-based method for partitioning alignments." Journal of Molecular Evolution 88 (8): 641-652.

[CT 6] Le Kim Thu, and Vinh Le Sy. 2021. "fastTIGER: A rapid method for estimating evolutionary rates of sites from large datasets." 2021 13th International Conference on Knowledge and Systems Engineering (KSE).

[CT 7] Le Kim Thu, and Vinh Le Sy. 2022. "A protein secondary structure-based algorithm for partitioning large protein alignments." 2022 14th International Conference on Knowledge and Systems Engineering (KSE) 1-5.

[CT 8] Le Kim Thu, Diep Hoang Thi, Dong Do Duc, Thang Bui Ngoc, Phuong Thao Nguyen Thi, and Vinh Le Sy. 2023. "gPartition: An Efficient Alignment Partitioning Program for Genome Datasets." VNU Journal of Science: Computer Science and Communication Engineering 29: 23-30.