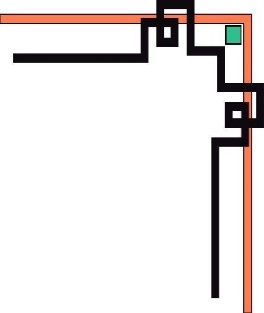
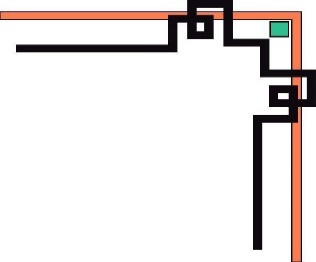
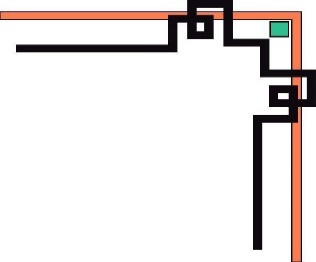
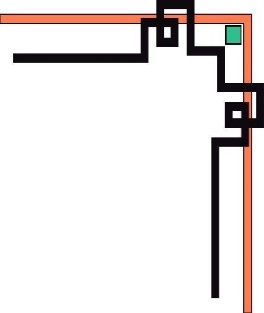
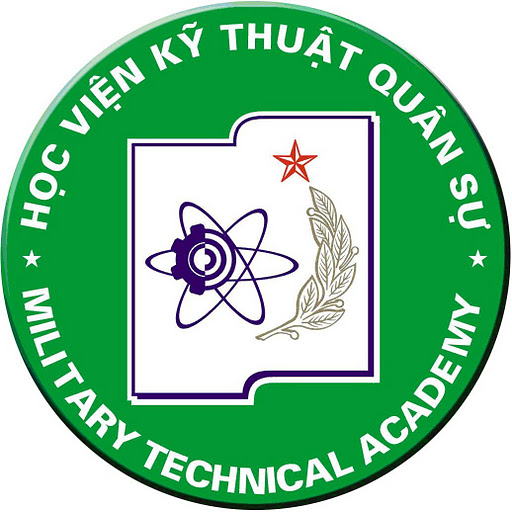
**HỌC VIỆN KỸ THUẬT QUÂN SỰ**



**KHOA CÔNG NGHỆ THÔNG TIN**

****

**BÁO CÁO**

**KHAI PHÁ DỮ LIỆU**

Đề tài: Ứng dụng LSTM vào dự báo sự biến động của mực nước sông theo thời gian

Sinh viên: Trần Trung Thiệp, Kiều Đức Long

Giáo viên hướng dẫn: Hồ Nhật Quang

Hà Nội, 10/2022

**Mục Lục**

[1. Mở đầu 2](#_Toc117975027)

[2. Phương pháp nghiên cứu 3](#_Toc117975028)

[**2.1. Phương pháp dự báo 4**](#_Toc117975029)

[**2.1.1. Phân tích dự báo 4**](#_Toc117975030)

[**2.1.2. Mô hình hóa dự báo 4**](#_Toc117975031)

[**2.1.3. Học có hướng dẫn (Supervised Learning – SL) 5**](#_Toc117975032)

[**2.1.3.1. Hồi quy tuyến tính (Linear Regression) 6**](#_Toc117975033)

[**2.1.3.2. Phân lớp (Classification) 8**](#_Toc117975034)

[**2.1.4. Học không cần hướng dẫn (Unsupervised Learning – USL) 10**](#_Toc117975035)

[**2.2. Mô hình LSTM 11**](#_Toc117975036)

[**2.3. Thu thập dữ liệu về khu vực nghiên cứu 14**](#_Toc117975037)

[**2.4. Phương pháp đánh giá 14**](#_Toc117975038)

[3. Kết quả nghiên cứu 15](#_Toc117975039)

[4. Kết luận 18](#_Toc117975040)

[Tài liệu tham khảo 19](#_Toc117975041)

1. **Mở đầu**

Diễn biến mực nước sông là một quá trình phức tạp, biến đổi theo không gian và thời gian.Việc dự báo chính xác mực nước là một trong những yêu cầu cấp bách nhằm giảm thiểu các rủi ro do lũ gây ra và có ý nghĩa quan trọng trong việc xây dựng phương án phòng, chống lũ. Các mô hình truyền thống được sử dụng để dự báo ở Việt Nam cũng như trên thế giới là các mô hình số về thủy lực và thủy văn. Các mô hình này yêu cầu một số lượng lớn các dữ liệu đầu vào như: đặc điểm lưu vực, địa hình, dự báo lượng mưa, quan hệ mưa – dòng chảy, quan hệ lưu lượng - mực nước theo thời gian tại một số vị trí.

Một trong những giải pháp hiệu quả cho việc dự báo là sử dụng mô hình Mạng thần kinh nhân tạo (Artificial Neural Network – ANN). Trên thế giới các mô hình ANN đã được sử dụng rộng rãi trong dự báo lũ từ những năm 1990 (Sung, J.Y.và các cộng sự, 2017). Cùng với đó, các nhà nghiên cứu đã áp dụng các thuật toán vào mô hình Mạng thần kinh để làm tăng độ chính xác của kết quả dự báo. Trong những năm gần đây, phương pháp Học sâu (Deep learning) dựa trên nền tảng là các mạng thần kinh nhân tạo đang phát triển rất nhanh và thu hút sự quan tâm của nhiều nhà khoa học. Có thể kể đến nghiên cứu của Chen, J.F. và các cộng sự (2014), ở đây thuật toán Cuckoo Search đã được áp dụng để dự báo dòng chảy đến hồ Hòa Bình, Việt Nam; Nguyễn Thanh Tùng (2016) đã sử dụng phương pháp Random Forest cũng để dự báo dòng chảy đến hồ này; Trương Xuân Nam và các cộng sự (2016) đã sử dụng phương pháp Học sâu để dự báo lưu lượng nước đến hồ Hòa Bình. Các nghiên cứu này đều có một điểm chung, đó là sử dụng dữ liệu của mùa kiệt với bước thời gian quan trắc 10 ngày làm đầu vào và đưa ra dự báo dòng chảy cho 10 ngày sau đó. Các kết quả nghiên cứu đều khẳng định khả năng tiềm ẩn của mô hình mạng thần kinh. Mô hình ANN cũng được áp dụng để dự báo mực nước ở trạm Hirakata, Nhật Bản (Kim, S. và các cộng sự, 2017). Nghiên cứu này chỉ sử dụng dữ liệu mực nước thực đo theo giờ ở các trạm thượng lưu để dự báo mực nước ở hạ lưu trước 3, 6 và 9 giờ. Kết quả đạt được cho thấy mô hình mạng thần kinh có thể áp dụng cho việc cảnh báo lũ trên sông. Wang, Y. và các cộng sự (2017) đã sử dụng mạng thần kinh LSTM để dự báo chất lượng nước cho hồ Taihu, Trung Quốc. Nghiên cứu chỉ ra rằng mô hình LSTM đưa ra dự báo chính xác hơn so với các mô hình mạng thần kinh khác.

Trong bài báo cáo này, chúng em sử dụng mô hình LSTM để dự báo mực nước sông mà không cần các thông tin dự báo lượng mưa, cũng như số liệu địa hình và tình hình sử dụng đất. Mô hình này được áp dụng để dự báo mực nước trước 1h, 7h, 13h, 19h tại mực nước của các trạm thủy văn tỉnh Bình Định năm 2022.

1. **Phương pháp nghiên cứu**

## 2.1. Phương pháp dự báo

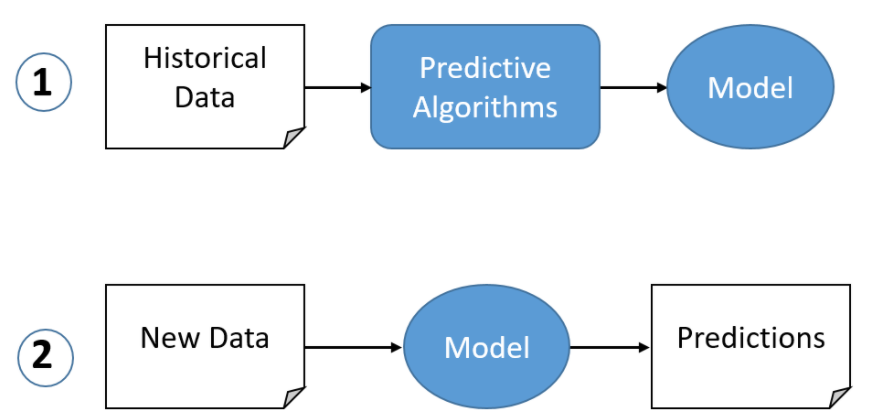
### 2.1.1. Phân tích dự báo

Phân tích dự báo **(predictive analytics)** đang là một hướng quan trọng trong lĩnh vực khai phá dữ liệu, nhất là khai phá dữ liệu lớn. Đây là nghiệp vụ trích chọn thông tin từ dữ liệu để phát hiện các mẫu hành vi, thói quen của đối tượng chủ thể, từ đó đưa ra dự báo về các hành vi và xu hướng trong tương lai của đối tượng đó.

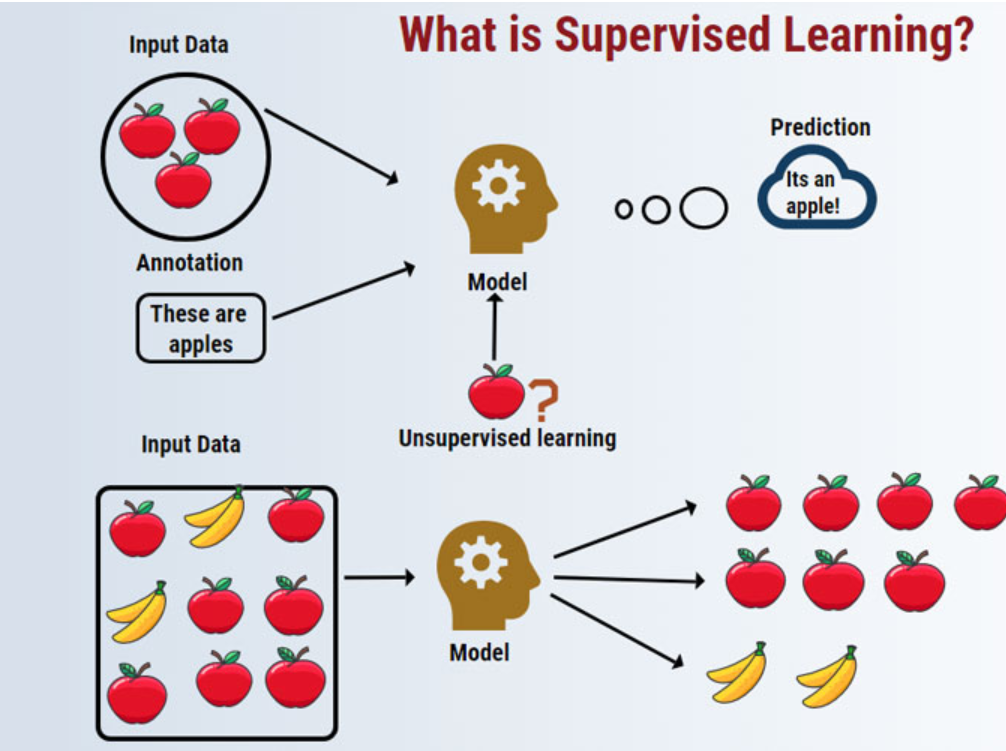
Các công cụ phân tích dự báo có thể giúp các công ty đánh giá trước được khả năng bán hàng, chẳng hạn dự báo một khách hàng nào đó sẽ mua hay không sản phẩm của mình, nếu mua thì khi nào và mua bao nhiêu. Dựa trên những thông tin này, các công ty có thể điều chỉnh chiến dịch tiếp thị và bán hàng của mình cho phù hợp.

### 2.1.2. Mô hình hóa dự báo

Mô hình dự đoán (Prediction Modeling) là một kỹ thuật thống kê thường được sử dụng để dự đoán hành vi trong tương lai. Các giải pháp mô hình dự đoán là một dạng công nghệ khai thác dữ liệu hoạt động (data mining) bằng cách phân tích dữ liệu lịch sử và hiện tại, đồng thời tạo ra một mô hình để giúp dự đoán kết quả trong tương lai.



### 2.1.3. Học có hướng dẫn (Supervised Learning – SL)



Chúng ta cần phải cung cấp cho máy tính một tập dữ liệu mấu (ví dụ như một tập hợp các câu hỏi và các câu trả lời đúng chẳng hạn). Sau đó chúng ta kì vòng rằng máy tính có thể thông qua các thuật toán, các mô hình được cài đặt để tìm ra các câu trả lời cho các câu hỏi chưa có đáp án trong tương lai.

Đó chính là tư tưởng của bài toán dự đoán (prediction) mà chúng ta sắp thảo luận dưới đây. Để làm được việc này người ta thường tạo ra một tập dữ liệu chuẩn để làm tập dữ liệu huấn luyện (training dataset). Trên tập này chúng ta sẽ “hướng dẫn” cho máy tính “học” thông qua các mô hình mà chúng ta quy định. Các tham số sinh ra được từ tập dữ liệu huẩn luyện sẽ được sử dụng để thẩm định lại tính thích hợp của mô hình trên tập dữ liệu kiểm tra (testing dataset). Có hai lớp bài toán điển hình trong SL đó là **hồi quy (regression)** và **phân lớp (Classification)**

#### 2.1.3.1. Hồi quy tuyến tính (Linear Regression)

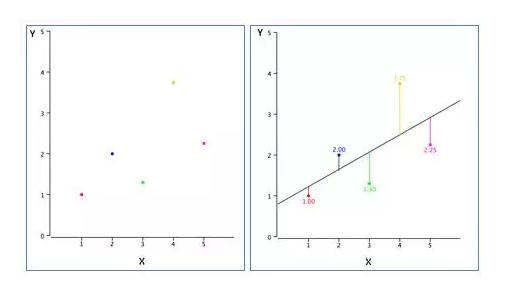
Là một phương pháp thống kê để hồi quy dữ liệu với biến phụ thuộc có giá trị liên tục trong khi các biến độc lập có thể có một trong hai giá trị liên tục hoặc là giá trị phân loại. Nói cách khác "Hồi quy tuyến tính" là một phương pháp để dự đoán biến phụ thuộc (Y) dựa trên giá trị của biến độc lập (X). Nó có thể được sử dụng cho các trường hợp chúng ta muốn dự đoán một số lượng liên tục. Ví dụ, dự đoán giao thông ở một cửa hàng bán lẻ, dự đoán thời gian người dùng dừng lại một trang nào đó hoặc số trang đã truy cập vào một website nào đó v.v...

**Một số khái niệm toán học về thống kê của hồi quy**:

* Tương quan (r) - Giải thích mối quan hệ giữa hai biến, giá trị có thể chạy từ -1 đến +1.
* Phương sai (σ2) - Đánh giá độ phân tán trong dữ liệu của bạn.
* Độ lệch chuẩn (σ) - Đánh giá độ phân tán trong dữ liệu của bạn (căn bậc hai của phương sai).
* Phân phối chuẩn.
* Sai số (lỗi) - {giá trị thực tế - giá trị dự đoán}.

**Đường hồi quy tuyến tính:**

Trong khi sử dụng hồi quy tuyến tính, mục tiêu của chúng ta là để làm sao một đường thẳng có thể tạo được sự phân bố gần nhất với hầu hết các điểm. Do đó làm giảm khoảng cách (sai số) của các điểm dữ liệu cho đến đường đó.



Ví dụ, ở các điểm ở hình trên (trái) biểu diễn các điểm dữ liệu khác nhau và đường thẳng (bên phải) đại diện cho một đường gần đúng có thể giải thích mối quan hệ giữa các trục x & y. Thông qua, hồi quy tuyến tính chúng ta cố gắng tìm ra một đường như vậy. Ví dụ, nếu chúng ta có một biến phụ thuộc Y và một biến độc lập X - mối quan hệ giữa X và Y có thể được biểu diễn dưới dạng phương trình sau:

**Y = B0 + B1\*X**

Trong đó:

* Y = Biến phụ thuộc
* X = biến độc lập
* Β0 = Hằng số
* Β1 = Hệ số mối quan hệ giữa X và Y

**Một vài tính chất của hồi quy**:

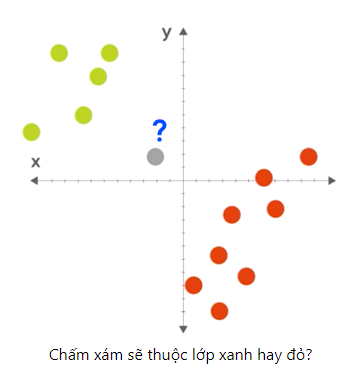
* Đường hồi quy luôn luôn đi qua trung bình của biến độc lập (x) cũng như trung bình của biến phụ thuộc (y)
* Đường hồi qui tối thiểu hóa tổng của "Diện tích các sai số". Đó là lý do tại sao phương pháp hồi quy tuyến tính được gọi là "Ordinary Least Square (OLS)"
* Β1 giải thích sự thay đổi trong Y với sự thay đổi X bằng một đơn vị. Nói cách khác, nếu chúng ta tăng giá trị của X bởi một đơn vị thì nó sẽ là sự thay đổi giá trị của Y.

#### 2.1.3.2. Phân lớp (Classification)

Bài toán phân lớp là quá trình phân lớp 1 đối tượng dữ liệu vào 1 hay nhiều lớp đã cho trước nhờ 1 mô hình phân lớp (model).

* Mô hình này được xây dựng dựa trên 1 tập dữ liệu được xây dựng trước đó có gán nhãn (hay còn gọi là tập huấn luyện).
* Quá trình phân lớp là quá trình gán nhãn cho đối tượng dữ liệu.

Bài toán phân lớp (classification) và bài toán gom cụm (cluster) là 2 bài toán lớn trong lĩnh vực Machine Learnig (ML).



Như vậy, nhiệm vụ của bài toán phân lớp là cần tìm 1 mô hình phần lớp để khi có dữ liệu mới thì có thể xác định được dữ liệu đó thuộc vào phân lớp nào.

Có nhiều bài toán phân lớp dữ liệu như phân lớp nhị phân (binary classification), phân lớp đa lớp (multiclass classification), phân lớp đa trị.

* Bài toán phân lớp nhị phân là bài toán gán nhãn dữ liệu cho đối tượng vào 1 trong 2 lớp khác nhau dựa vào việc dữ liệu đó có hay không có các đặc trưng (feature) của bộ phân lớp.
* Bài toán phân lớp đa lớp là quá trình phân lớp dữ liệu với số lượng lớp lớn hơn 2. Như vậy với từng dữ liệu phải xem xét và phân lớp chúng vào những lớp khác nhau chứ không phải là 2 lớp như bài toán phân lớp nhị phân. Và thực chất bài toán phân lớp nhị phân là 1 bài toán đặt biệt của phân lớp đa lớp.

Ứng dụng của bài toán này được sử dụng rất nhiều và rộng rãi trong thực tế ví dụ như bài toán nhận dạng khuôn mặt, nhận diện giọng nói, phát hiện email spam,…



Phân lớp nhận dạng email spam

Và đối với các bài toán phân lớp dữ liệu sử dụng các thuật toán học có giám sát (supervised learning) để xây dựng mô hình cho bài toán này.

**Quá trình phân lớp dữ liệu:**

Để xây dựng được mô hình phân lớp và đánh giá được mô hình phải trải qua các quá trình sau.

**1. Chuẩn bị tập dữ liệu huấn luyện (dataset) và rút trích đặc trưng (feature extraction)**

Công đoạn này được xem là công đoạn quan trọng trong các bài toán về Machine Learning. Vì đây là input cho việc học dể tìm ra mô hình của bài toán. Chúng ta phải biết cần chọn ra những đặt trưng tốt (good feature) của dữ liệu, lược bỏ những đặc trưng không tốt của dữ liệu, gây nhiễu (noise). Uớc lượng số chiều của dữ liệu bao nhiêu là tốt hay nói cách khác là chọn bao nhiêu feature. Nếu số chiều quá lớn gây khó khăn cho việc tính toán thì phải giảm số chiều của dữ liệu nhưng vẫn giữ được độ chính xác của dữ liệu (reduce demension).

Ở bước này chúng ta cũng chuẩn bị bộ dữ liệu để test trên mô hình. Thông thường sẽ sử dụng cross-validation (kiểm tra chéo) để chia tập datasets thành 2 phần, 1 phần phục vụ cho training (**training datasets**) và phần còn lại phục vụ cho mục đích testing trên mô hình (**testing dataset).** Có 2 cách thường sử dụng trong cross-validation là **splitting** và **k-fold**.

**2. Xây dựng mô hình phân lớp (classifier model)**

Mục đích của mô hình huấn luyện là tìm ra hàm f(x) và thông qua hàm f tìm được để gán nhãn cho dữ liệu, bước này thường được gọi là learning hay training.

**f(x) = y**

* x: các feature hay input đầu vào của dữ liệu.
* y: nhãn lớp hay output đầu ra.

Thông thường để xây dựng mô hình phân lớp cho bài toán này cần sử dụng các thuật toán học giám sát (supervised learning) như k-nearest neighbors, Neural Network, SVM, Decision tree, Naive Bayes.

**3. Kiểm tra dữ liệu với mô hình (making predictions)**

Sau khi đã tìm được mô hình phân lớp ở bước 2, thì ở bước này sẽ đưa vào các dữ liệu mới để kiểm tra trên mô hình phân lớp.

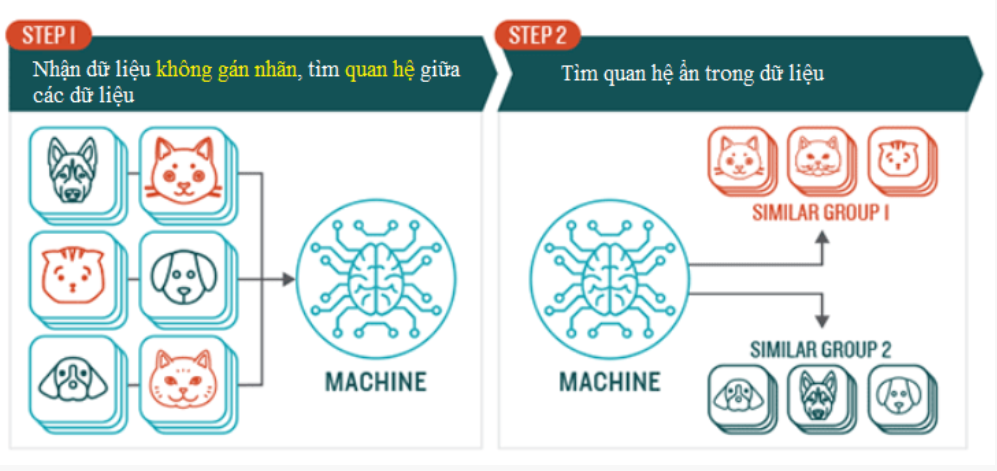
**4. Đánh giá mô hình phân lớp và chọn ra mô hình tốt nhất**

Bước cuối cùng sẽ đánh giá mô hình bằng cách đánh giá mức độ lỗi của dữ liệu testing và dữ liệu training thông qua mô hình tìm được. Nếu không đạt được kết quả mong muốn thì phải thay đổi các tham số (turning parameters) của các thuật toán học để tìm ra các mô hình tốt hơn và kiểm tra, đánh giá lại mô hình phân lớp, và cuối cùng chọn ra mô hình phân lớp tốt nhất cho bài toán.

### 2.1.4. Học không cần hướng dẫn (Unsupervised Learning – USL)

Nếu như học có hướng dẫn chúng ta cần phải cung cấp cho máy tính một tập dữ liệu mẫu với các câu trả lời đúng cho từng câu hỏi (hướng dẫn) thì có những thuật toán trong USL không cần làm điều đó. Các thuật toán USL có khả năng khai thác cấu trúc ẩn của dữ liệu từ một tập dữ liệu chưa được gán nhãn trước (chưa có câu trả lời mẫu). Các hướng tiếp cận điển hình trong USL có thể kể đến như Phân cụm, Phát hiện bất thường, Mạng Nơ ron nhân tạo.

Học không giám sát là một lớp mô hình học sử dụng một thuật toán để mô tả hoặc trích xuất ra các mối quan hệ tiềm ẩn trong dữ liệu. Khác với học có giám sát, học không giám sát chỉ thực thi trên dữ liệu đầu vào không cần các thuộc tính nhãn, hoặc mục tiêu của việc học. Tức là không hề được cung cấp trước một kiến thức nào trước trừ dữ liệu. Các dữ liệu không được “hướng dẫn” trước như trong trường hợp học có giám sát. Các thuật toán cần học được từ dữ liệu mà không hề có bất cứ sự hướng dẫn nào.



## 2.2. Mô hình LSTM

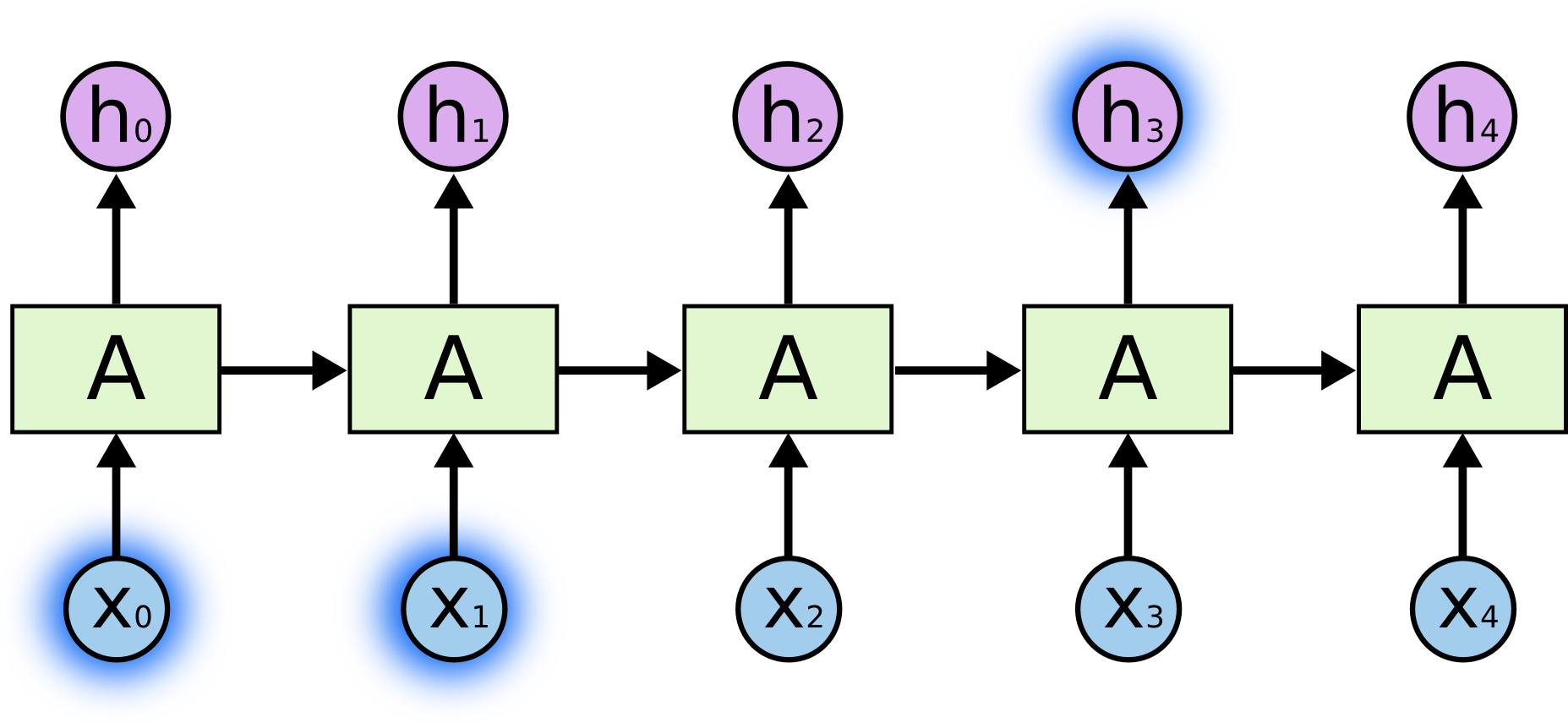
**2.2.1. Đánh giá LSTM**

LSTM là một mạng cải tiến của RNN nhằm giải quyết vấn đề nhớ các bước dài của RNN.

* **Vấn đề phụ thuộc xa:**

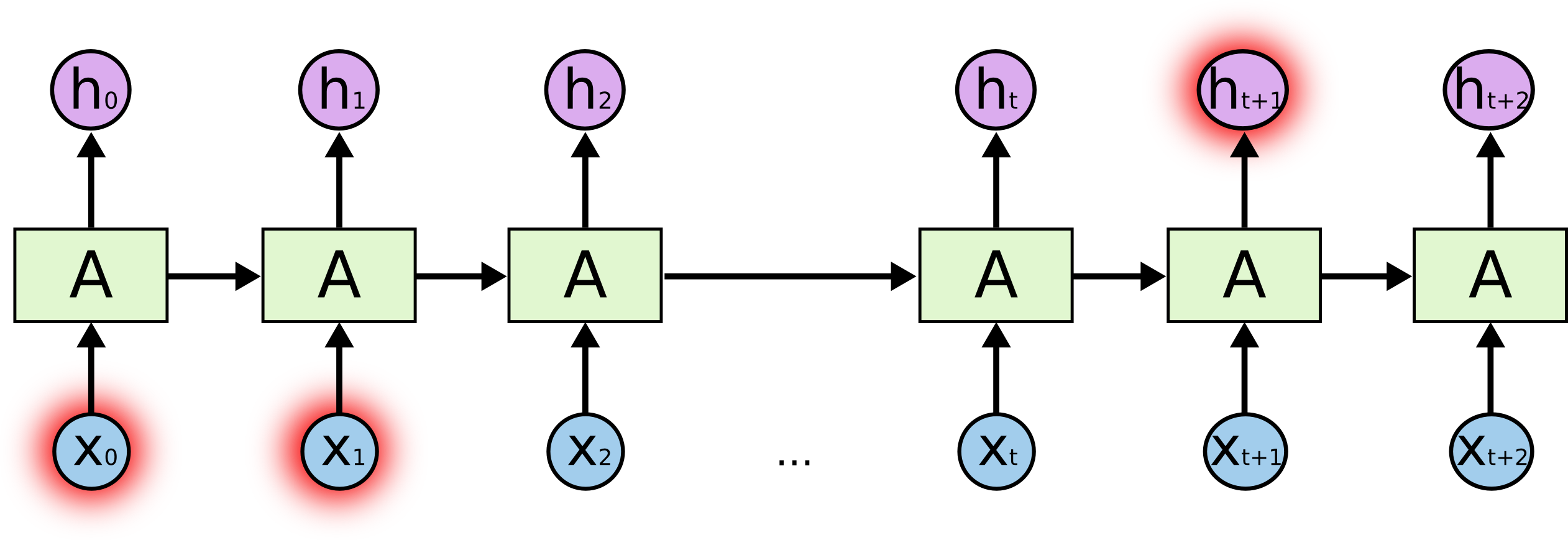
Một điểm nổi bật của RNN chính là ý tưởng kết nối các thông tin phía trước để dự đoán cho hiện tại. Việc này tương tự như ta sử dụng các cảnh trước của bộ phim để hiểu được cảnh hiện thời. Nếu mà RNN có thể làm được việc đó thì chúng sẽ cực kì hữu dụng, tuy nhiên liệu chúng có thể làm được không? Câu trả lời là còn tùy.

Đôi lúc ta chỉ cần xem lại thông tin vừa có thôi là đủ để biết được tình huống hiện tại. Ví dụ, ta có câu: “các đám may trên bầu trời” thì ta chỉ cần đọc tới “các đám mây trên bầu” là đủ biết được chữ tiếp theo là “trời” rồi. Trong tình huống này, khoảng cách tới thông tin có được cần để dự đoán là nhỏ, nên RNN hoàn toàn có thể học được.



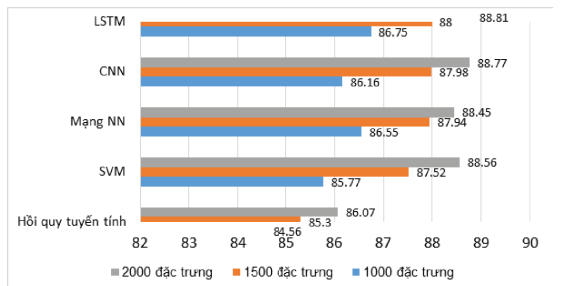
Nhưng trong nhiều tình huống ta buộc phải sử dụng nhiều ngữ cảnh hơn để suy luận. Ví dụ, dự đoán chữ cuối cùng trong đoạn: “I grew up in France… I speak fluent French.”. Rõ ràng là các thông tin gần (”I speak fluent”) chỉ có phép ta biết được đằng sau nó sẽ là tên của một ngôn ngữ nào đó, còn không thể nào biết được đó là tiếng gì. Muốn biết là tiếng gì, thì ta cần phải có thêm ngữ cảnh “I grew up in France” nữa mới có thể suy luận được. Rõ ràng là khoảng cách thông tin lúc này có thể đã khá xa rồi.

Thật không may là với khoảng cách càng lớn dần thì RNN bắt đầu không thể nhớ và học được nữa.



LSTM được thiết kế để tránh được vấn đề phụ thuộc xa (long-term dependency). Việc nhớ thông tin trong suốt thời gian dài là đặc tính mặc định của chúng, chứ ta không cần phải huấn luyện nó để có thể nhớ được. Tức là ngay nội tại của nó đã có thể ghi nhớ được mà không cần bất kì can thiệp nào.

* **Ưu, nhược điểm của LSTM**
* **Ưu điểm:**
* Không cần phải huấn luyện nó để có thể nhớ được, Do đó, giải quyết được vấn đề phụ thuộc xa mà RNN không làm được.
* Kết quả dự báo có tỉ lệ chính xác cao hơn.

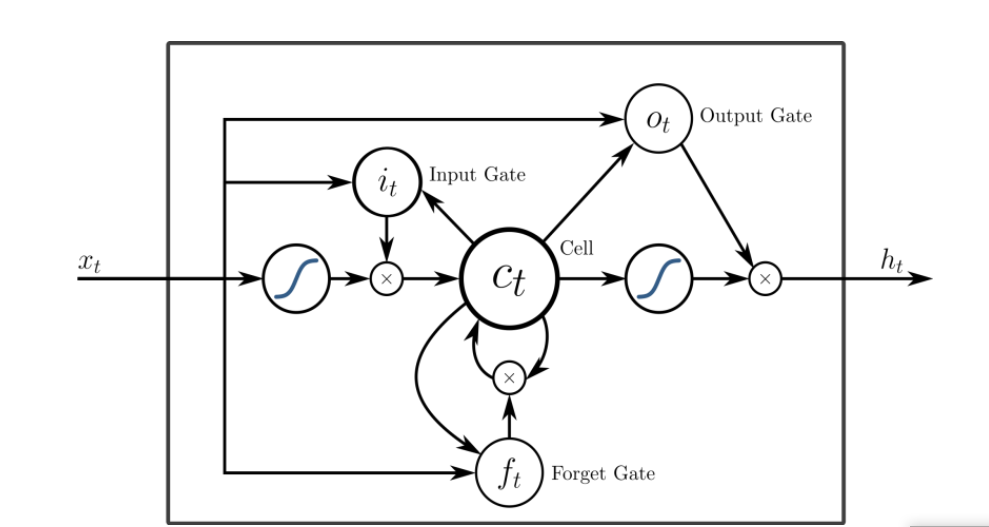


* **Nhược điểm:**
* Phức tạp hơn mạng RNN do đó tốc độ thường chậm hơn.
* Do có một tập các công thức, khiến cho LSTM trở nên khá khó hiểu.

Mô hình đề xuất dựa trên mô hình mạng thần kinh sâu LSTM, đây là một dạng đặc biệt của RNN (Recurrent Neural Network - Mạng thần kinh hồi quy). LSTM được giới thiệu bởi Hochreiter và Schmidhuber (1997) nhằm giải quyết các bài toán về phụ thuộc xa (long-term dependency).

Dữ liệu mực nước là một dạng dữ liệu chuỗi thời gian, dữ liệu mực nước ở thời điểm t phụ thuộc vào mực nước ở thời điểm t-1, t-2,… Để dự báo mực nước, mạng nơron hồi quy (Recurrent Neural Networks -RNN) [11] được sử dụng thay cho mạng nơron thông thường. Mạng nơron hồi quy là mạng có trạng thái bên trong được đưa trở lại đầu vào. Nghĩa là mạng tính toán đầu ra bằng thông tin hiện tại và dự đoán của đầu vào trước đó. Trong RNN các đầu vào sẽ được kết hợp với phần tử của lớp ẩn trước đó để tính ra giá trị phần tử của lớp ẩn hiện tại sau đó tính đầu ra hiện tại.

Như vậy kết quả từ các quá trình tính toán trước đã được “nhớ” bằng cách kết hợp thêm phần tử ẩn trước đó để tăng độ chính xác cho những dự đoán ở hiện tại. Tuy nhiên, mô hình này gặp khó khăn liên quan đến việc cập nhật trọng số trong lớp ẩn, vì mất mát đạo hàm xảy ra dẫn đến giá trị gradient sẽ ngày càng gần với giá trị 0. Để giải quyết vấn này, một dạng mở rộng của RNN, mô hình LSTM (Long short-term memory được đề xuất bởi Hochreiter & Schmidhuber (1997) [10]. Mô hình LSTM này được xem là một giải pháp tối ưu hơn các mạng nơron truyền thống khác khi xử lý các vấn đề liên quan dự đoán chuỗi thời gian. Mỗi một nơron trong mô hình LSTM bao gồm ba cổng: cổng quên (forget gate), cổng vào (input gate) và cổng ra (output gate) mô tả như Hình 3. Các cổng là nơi sàng lọc thông tin đi qua nó, chúng được kết hợp bởi một tầng mạng sigmoid và một phép nhân. Tầng sigmoid sẽ cho đầu ra là một số trongkhoảng [0, 1], mô tả có bao nhiêu thông tin có thể được thông qua. Khi đầu ra là 0 thì có nghĩa là không cho thông tin nào qua cả, còn khi là 1 thì có nghĩa là cho tất cả các thông tin đi qua nó. Cổng (gate) trong LSTM có tác dụng loại bỏ hay thêm vào các thông tin cần thiết cho trạng thái tế bào. Cổng ra (output gate) quyết định số lượng nội dung trong ô nhớ sẽ hiển thị đầu ra của khối. Nội dung hiển thị này phụ thuộc vào thông tin từ cổng vào (input gate) và cổng quên (forget gate).



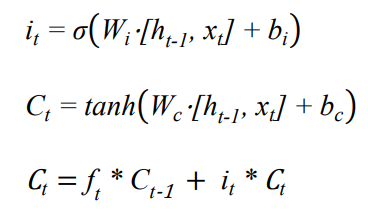
Hình 3: Cấu trúc của một LSTM nơron

Bước đầu tiên của mô hình LSTM là quyết định thông tin nào cần loại bỏ từ trạng thái tế bào. Quyết định này được đưa ra bởi cổng quên (forget gate). Đầu vào cho bước này là ht-1 (giá trị đầu ra tại thời điểm t-1) và xt (dữ liệu đầu vào), đầu ra ft là một số trong khoảng từ [0, 1] cho mỗi số trong trạng thái tế bào Ct-1. Công thức của cổng quên (forget gate) được tính như sau:

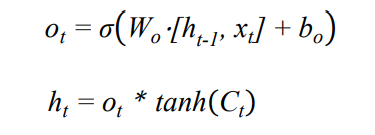


Trong đó: là hàm sigmoid, Wf và bf lần lượt là trọng số và bias của cổng quên (forget gate)

Bước tiếp theo là quyết định xem thông tin mới nào sẽ lưu vào trạng thái tế bào. Việc này gồm 2 phần, đầu tiên là sử dụng cổng vào (input gate) để quyết định giá trị nào sẽ cần cập nhật. Tiếp theo là sử dụng hàm tanh tạo ra một véctơ cho giá trị mới Ct nhằm thêm vào cho trạng thái. Trong bước tiếp theo, ta sẽ kết hợp 2 giá trị đó lại để tạo ra một cập nhật cho trạng thái. Công thức thực hiện cổng vào và cập nhật trạng thái tế bào:



Trong đó: là hàm sigmoid, Wi và bi lần lượt là trọng số và bias của cổng vào (input gate), Wc và bc lần lượt là trọng số và bias của trạng thái tế bào, Ct-1 và Ct là trạng thái tế bào lần lượt ở thời điểm t-1 và t. Ở bước cuối cùng, giá trị đầu ra (ht) sẽ được quyết định bởi trạng thái của tế bào muốn xuất ra (output gate, nhưng sẽ được tiếp tục sàng lọc. Đầu tiên, cần chạy một hàm sigmoid để quyết định phần nào của trạng thái tế bào muốn xuất ra. Sau đó, đưa trạng thái tế bào qua một hàm tanh để co giá trị nó về khoảng [-1, 1] và nhân nó với đầu ra của hàm sigmoid để được giá trị đầu ra mong muốn.

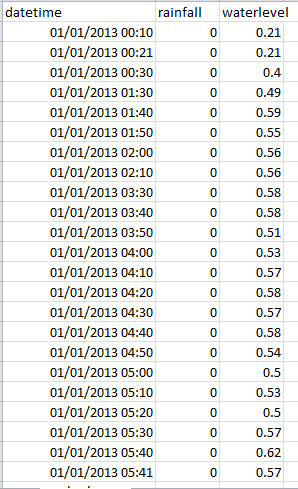


Trong đó: là hàm sigmoid, Wo và bo lần lượt là trọng số và bias của cổng ra (output gate)

Chức năng bộ nhớ có chọn lọc của LSTM được thực hiện bởi cơ chế cổng (gate) giúp LSTM phù hợp hơn để xử lý các vấn đề liên quan đến dự đoán chuỗi thời gian so với các mạng nơron truyền thống khác. Mô hình dự báo mực nước sông dựa trên. LSTM có thể tận dụng tối đa chuỗi thời gian của thông tin mực nước đo được tại các trạm để cải thiện độ chính xác của mô hình dự đoán.

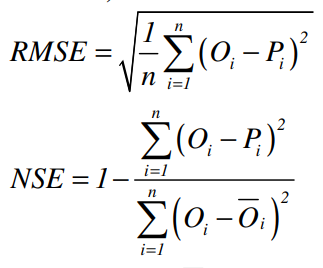
## 2.3. Thu thập dữ liệu về khu vực nghiên cứu

Dữ liệu được thu thập bao gồm: các số liệu mực nước theo giờ tại các trạm thủy văn của tỉnh Bình Định từ 01/2013 – 12/2015**.**



## 2.4. Phương pháp đánh giá

Để đánh giá hiệu quả của mô hình dự báo, chúng em sử dụng hai trị số, đó là RMSE (Root Mean Squared Error – sai số căn quân phương) và NSE (Nash Sutcliffe Efficiency – hệ số Nash)



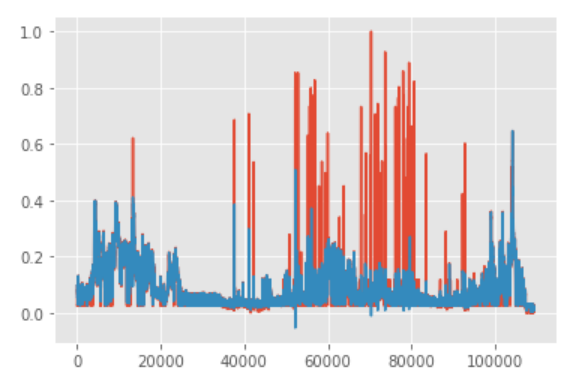
Trong đó: Oi , Oi và Pi lần lượt là giá trị thực đo, giá trị thực đo trung bình và giá trị dự báo của mẫu thứ i tương ứng. Mô hình dự báo cho kết quả tốt nếu RMSE nhỏ và NSE lớn.

1. **Kết quả nghiên cứu**

Mô hình LSTM được xây dựng để dự báo mực nước lần lượt cách nhua mỗi lần đo là 30 phút. Với mỗi trường hợp, các thông số của mô hình có thể được điều chỉnh bằng cách thử dần cho đến khi đạt kết quả tốt nhất. Mô hình cũng được lập trình để tự động tính các trị số của RMSE và NSE theo công thức phía trên. Sau quá trình hiệu chỉnh và kiểm định, các thông số phù hợp nhất của mô hình cho mỗi trường hợp đã được lựa chọn để phục vụ việc dự báo.

Sử dụng cấu trúc mô hình đã được lựa chọn như trên, với thời gian dự báo cách nhau 30 phút, bằng cách so sánh kết quả dự báo với số liệu thực đo mực nước tại An hòa trong thời gian 3 năm, từ 01/2013 đến 12/2015. Kết quả kiểm định mô hình được thể hiện trên các đồ thị bên dưới.

Kết quả dự báo mực nước tại trạn **An hòa** của tỉnh **Bình Định** trong 3 năm.



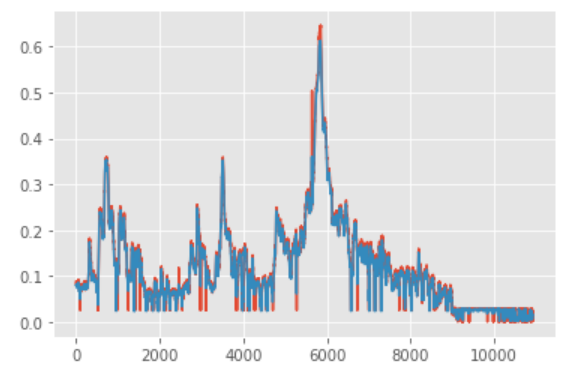
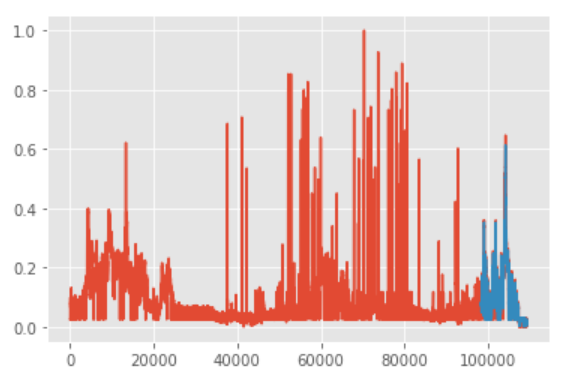
**Trong đó:**

* Đường màu đỏ: là kết quả đo được
* Đường màu xanh: là kết quả dự báo
* Sử dụng Epoch = 1000 cho thực nghiệm

RMSE (sai số căn quân phương):



Sử dụng đo thực từ tháng 9/2022 để dự báo mực nước tháng 10/2022, kết quả hình bên dưới:



**Trong đó:**

* Đường màu đỏ: là kết quả đo được
* Đường màu xanh: là kết quả dự báo
* Sử dụng Epoch = 1000 cho thực nghiệm

RMSE (sai số căn quân phương):



1. **Kết luận**

Có thể thấy rằng, không có quy tắc cụ thể nào cho việc lựa chọn các thông số của mô hình như hệ số học, số lượng chuỗi, số lượng lớp ẩn, số lượng Epoch cũng như cấu trúc của mô hình. Việc lựa chọn các thông số này dựa vào quá trình thử và đánh giá sai số. Các thông số tốt nhất đã được chọn và đánh giá qua quá trình hiệu chỉnh và kiểm định mô hình.

Mạng thần kinh sâu có ưu điểm là đơn giản hơn so với các mô hình thủy văn, thủy lực, đặc biệt nó có thể dự báo cho vùng bị ảnh hưởng của thủy triều trong trường hợp thiếu số liệu địa hình, khi mà không thể sử dụng các mô hình thủy lực, thủy văn. Lúc đó, mô hình mạng thần kinh LSTM là giải pháp tốt cho việc dự báo mực nước theo thời gian thực, thay thế các mô hình đã có. Vì vậy, có thể ứng dụng mô hình này để cảnh báo lũ trên các sông suối của Việt Nam.

# Tài liệu tham khảo

Trương Xuân Nam và Nguyễn Thanh Tùng (2016). Deep learning: Ứng dụng cho dự báo lưu lượng nước đến hồ chứa hòa bình. Hội nghị khoa học thường niên, Đại học Thủy lợi.

Viện Kỹ thuật tài nguyên nước (2011). Quy hoạch thủy lợi chống ngập úng thành phố Hải Phòng. Trường Đại học Thủy lợi.

Chen, J.F., Hsieh, H.N., and Do, Q.H. (2014). Forecasting Hoabinh Reservoir’s Incoming Flow: An Application of Neural Networks with the Cuckoo Search Algorithm. Information. 5, 570-586. Kim, S. and Tachikawa, Y. (2017). Real-time river-stage prediction with artificial neural network based on only upstream observation data. Annual Journal of Hydraulic Engineering, JSCE, Vol. 61.

Nguyen , T.T. (2015). An l1-regression Random Forests Method For Forecasting of Hoa Binh Reservoir’s Incoming Flow. International Conference on Intelligent Systems and Knowledge Engineering.

Olah, C.(2015). Understanding LSTM Networks. GITHUB blog, Retrieved from http://colah.github.io/posts/2015-08-Understanding-LSTMs/. Posted on August 27, 2015.

Sung, J.Y., Lee, J., Chung, I.M., and Heo, J.H. (2017). Hourly Water Level Forecasting at Tributary Affected by Main River Condition. KSCE Journal of Civil Engineering. 9, 644.

Wang, Y., Zhou, J., Chen. K., Wang, Y., and Liu. L. (2017). Water Quality Prediction Method Based on LSTM Neural Network. International Conference on Intelligent Systems and Knowledge Engineering.