**TRƯỜNG ĐẠI HỌC MỎ - ĐỊA CHẤT**

**A blue circle with white text and a globe

Description automatically generatedKHOA CÔNG NGHỆ THÔNG TIN**

**Bài tập giữa kỳ**

**Nhập Môn Mô Hình Cơ Sở Hạ Tầng Dữ Liệu Địa Không Gian**

|  |  |
| --- | --- |
| **Sinh Viên Thực Hiện** | **Cán Bộ Hướng Dẫn** |
| **Trần Quang Vinh:2121060564**  **Lê Minh Chiến:2121050728**  **Nguyễn Đình Trung:** **2121050532**  **Phan Việt Đức:** **2121051124**  **Nguyễn Hữu Dũng:2121051403** | **Ts: Trần Mai Hương** |

Hà Nội, Ngày 12 Tháng 12 Năm 2024

MỤC LỤC

[**LỜI MỞ ĐẦU** 4](#_Toc184853188)

[**CHƯƠNG 1 : TỔNG QUAN VỀ LĨNH VỰC NGHIÊN CỨU** 5](#_Toc184853189)

[**I, Tổng quan** 5](#_Toc184853190)

[**1.1 Mục tiêu của đề tài** 5](#_Toc184853191)

[**1.2 Giới thiệu về dữ liệu** 5](#_Toc184853192)

[**1.3 Giới thiệu về các mô hình** 6](#_Toc184853193)

[**1.3.1 Giới thiệu về mô hình DNN** 6](#_Toc184853194)

[**1.3.2 Giới thiệu về mô hình RNN** 7](#_Toc184853195)

[**1.3.3 Giới thiệu về LSTM** 9](#_Toc184853196)

[**1.3.5 Giới thiệu về GRU** 11](#_Toc184853197)

[Hình 1(Công thức của RNN) 8](#_Toc184853064)

[Hình 2(Công thức tính tạo ra đầu ra tại thời điểm t) 8](#_Toc184853065)

[Hình 3(Công thức LSTM) 10](#_Toc184853066)

[Hình 4 Công thức GRU 12](#_Toc184853067)

# **LỜI MỞ ĐẦU**

Chuỗi thời gian đang được sử dụng như một công cụ hữu hiệu để phân tích trong kinh tế, xã hội cũng như trong nghiên cứu khoa học. Nghiên cứu dự báo chuỗi thời gian luôn là một bài toán gây được sự chú ý của các nhà toán học, kinh tế, xã hội học,... Các quan sát trong thực tế thường được thu thập dưới dạng chuỗi số liệu. Làm sao để từ chuỗi số liệu khó khăn, ta có thể tìm ra được một mô hình hay một quy luật nào đó của một quá trình có đủ cơ sở chính xác để phản ánh được chân thực dữ liệu đã có (kiểm tra) đồng thời lại có thể dự đoán cho những thời điểm trong tương lai chưa xảy ra?! Mà việc dự đoán được tương lai ra sao có lẽ luôn là những mong đợi thường trực của xã hội loài người.

Chính do tầm quan trọng của việc phân tích chuỗi thời gian như vậy, rất nhiều tác giả đã nghiên cứu và đề xuất các công cụ để phân tích thời chuỗi thời gian như sử dụng các công cụ thống kê hồi qui, phân tích Furie, mô hình ARIMA của Box-Jenkins,... Sau này có nhiều người sử dụng mạng Nơron để xử lý tính chất phi tuyến của chuỗi số liệu, có thể tìm thấy trong những cuốn sách chuyên khảo về vấn đề này, thí dụ như cuốn của Mandic và Chambers “Recurrent neural network and prediction” in vào năm 2001. Một hướng đi khác là sử dụng khái niệm mới để đưa ra thuật ngữ “Chuỗi thời gian mở”. Phương pháp sử dụng chuỗi thời gian mở đã được đưa ra từ năm 1994 và đến nay vẫn đang được tiếp tục nghiên cứu để làm tăng độ chính xác của dự báo...

Nghiên cứu chuỗi thời gian đòi hỏi một kiến thức rộng lớn về Thống kê, xác suất, trong kinh tế và khoa học. Các phép tính chủ yếu dựa trên các con số dữ liệu rời rạc, liên tục vì thế đòi hỏi phải có những thuật toán hay những phần mềm tính toán cho xác suất và thống kê chuyên dụng. Có nhiều phần mềm như thể như Eview, S-plus, R, v.v...

Chính vì thế, Luận văn tập trung đi vào 6 chương đầu trong cuốn “Chuỗi thời gian – Các ứng dụng trong tài chính với R và S-plus“ – *Time Series. Applications to Finance with R and S – Plus (R)* của tác giả Ngai Hai Chan – The University of Hong Kong. Bao gồm:

* **Chương 1**: Giới thiệu về chuỗi thời gian
* **Chương 2**: Giới thiệu về Lý thuyết xác suất của quá trình ngẫu nhiên
* **Chương 3**: Giới thiệu về mô hình trung bình trượt tự hồi quy
* **Chương 4**: Bàn về các Ước lượng trong miền thời gian
* **Chương 5**: Trình bày về 2 ví dụ cụ thể sử dụng phần mềm R cho những tính toán được trình bày ở các chương trước
* **Chương 6**: Trình bày về Dự đoán.

Luận văn này được hoàn thành dưới sự hướng dẫn tận tình của PGS. TS Phan Viết Thư, tác giả xin bày tỏ lòng biết ơn chân thành của mình đến với thầy! Tác giả cũng chân thành cảm ơn các thầy giáo trong trường ĐH Khoa học Tự nhiên – Đại học Quốc Gia Hà Nội đã tham gia giảng dạy và giúp đỡ em trong suốt quá trình học tập nâng cao trình độ kiến thức tại trường trong 2 năm học Cao học. Tuy nhiên, vì điều kiện thời gian và khả năng có hạn nên luận văn không thể tránh khỏi những thiếu sót. Tác giả kính mong các thầy cô và các bạn góp ý kiến để đề tài được hoàn thiện hơn.

# **CHƯƠNG 1 : TỔNG QUAN VỀ LĨNH VỰC NGHIÊN CỨU**

## **I, Tổng quan**

**1.1 Mục tiêu của đề tài**

* **Đánh giá công suất tiêu thụ năng lượng hiện tại**: Phân tích mức tiêu thụ năng lượng tại các khu vực dân cư, công nghiệp và thương mại trong thành phố Tétouan, từ đó đưa ra các số liệu cụ thể về nhu cầu điện và năng lượng của thành phố.
* **Xác định các yếu tố ảnh hưởng đến công suất tiêu thụ năng lượng**: Nghiên cứu các yếu tố như dân số, phát triển kinh tế, cơ cấu ngành nghề, và thói quen tiêu thụ của người dân có tác động đến mức độ tiêu thụ năng lượng.
* **Phân tích tác động của sự phát triển đô thị và công nghiệp**: Đánh giá sự gia tăng nhu cầu năng lượng do quá trình đô thị hóa, mở rộng các khu công nghiệp và dịch vụ trong thành phố.
* **Đề xuất giải pháp cho việc quản lý và sử dụng năng lượng hiệu quả**: Dựa trên các kết quả nghiên cứu, đề tài sẽ đưa ra các giải pháp giúp Tétouan phát triển một hệ thống năng lượng bền vững, bao gồm việc sử dụng năng lượng tái tạo và cải thiện hiệu quả sử dụng năng lượng trong các ngành công nghiệp và hộ gia đình.
* **Đánh giá các cơ hội và thách thức trong việc chuyển đổi sang nguồn năng lượng tái tạo**: Phân tích khả năng ứng dụng các nguồn năng lượng tái tạo (như năng lượng gió và năng lượng mặt trời) trong thành phố, và những khó khăn, thuận lợi của quá trình chuyển đổi này.

### **1.2 Giới thiệu về dữ liệu**

Bộ dữ liệu này ghi lại các thông tin chi tiết về điều kiện môi trường và mức tiêu thụ năng lượng của ba khu vực trong một khoảng thời gian cụ thể. Dữ liệu bao gồm các yếu tố khí hậu như nhiệt độ, độ ẩm, và tốc độ gió, cũng như các thông số về dòng năng lượng khuếch tán. Đây là những yếu tố có thể ảnh hưởng trực tiếp hoặc gián tiếp đến mức tiêu thụ năng lượng trong các khu vực thành phố Tétouan.

### **1.3 Giới thiệu về các mô hình**

#### **1.3.1 Giới thiệu về mô hình DNN**

* Định nghĩa : Mô hình Deep Neural Networks (DNN), hay Mạng Nơ-ron Sâu, là một dạng của mạng nơ-ron nhân tạo (Artificial Neural Network - ANN) với nhiều lớp ẩn (hidden layers). DNN là một phương pháp học máy mạnh mẽ, được áp dụng rộng rãi trong các lĩnh vực như nhận dạng hình ảnh, nhận dạng giọng nói, phân tích ngữ nghĩa trong văn bản và nhiều ứng dụng trí tuệ nhân tạo khác.
* Cấu trúc của DNN : DNN bao gồm ba thành phần chính là lớp đầu vào (Input Layer), Lớp ẩn (Hidden Layers), Lớp đầu ra (Output Layer). Lớp tiếp nhận dữ liệu đầu vào (ví dụ: hình ảnh, văn bản, tín hiệu âm thanh). Mỗi nút trong lớp đầu vào đại diện cho một đặc trưng của dữ liệu. DNN có một hoặc nhiều lớp ẩn giữa lớp đầu vào và lớp đầu ra. Các lớp này chịu trách nhiệm trích xuất các đặc trưng từ dữ liệu đầu vào thông qua các phép toán phi tuyến tính. Mỗi lớp ẩn có một số nút (neuron) liên kết với các nút của lớp trước và lớp sau. Các trọng số (weights) của kết nối giữa các nút được điều chỉnh trong quá trình huấn luyện. Lớp này cung cấp kết quả cuối cùng của mô hình. Ví dụ, trong bài toán phân loại, lớp đầu ra sẽ cung cấp xác suất cho từng lớp phân loại.
* Hoạt động của DNN là Lan truyền trước (Forward Propagation): Dữ liệu đầu vào được đưa qua các lớp của mạng nơ-ron. Mỗi lớp sẽ áp dụng một hàm kích hoạt (activation function) để xử lý và truyền tín hiệu đến lớp kế tiếp. Cuối cùng, lớp đầu ra sẽ cung cấp dự đoán của mô hình. Lan truyền ngược (Backpropagation): Khi mô hình đưa ra dự đoán, sự sai lệch (error) giữa dự đoán và giá trị thực tế sẽ được tính toán. Backpropagation sử dụng thuật toán Gradient Descent để điều chỉnh các trọng số của mạng sao cho sai số giảm xuống. Quá trình này lặp đi lặp lại cho đến khi mô hình đạt được độ chính xác tối ưu.
* Hàm kích hoạt là ReLU (Rectified Linear Unit): Một hàm kích hoạt rất phổ biến trong DNN, giúp giải quyết vấn đề biến mất gradient và cải thiện hiệu suất của mạng nơ-ron. Sigmoid: Hàm này cho kết quả từ 0 đến 1, thường được sử dụng trong các bài toán phân loại nhị phân. Tanh (Hyperbolic Tangent): Hàm này đưa giá trị đầu ra vào khoảng -1 đến 1, thường được sử dụng trong các mạng nơ-ron truyền thống.
* Ưu điểm của DNN là Khả năng học sâu: Với nhiều lớp ẩn, DNN có thể học và trích xuất các đặc trưng phức tạp từ dữ liệu, giúp cải thiện khả năng phân loại và dự đoán. Tính linh hoạt: DNN có thể áp dụng cho nhiều loại dữ liệu khác nhau như hình ảnh, âm thanh, văn bản, và dữ liệu thời gian. Hiệu quả trong các bài toán lớn: DNN đặc biệt hiệu quả trong việc xử lý dữ liệu lớn, nhờ vào khả năng trích xuất và học các đặc trưng sâu từ dữ liệu.
* Ứng dụng của DNN là Nhận dạng hình ảnh và video: Phân loại hình ảnh, nhận diện vật thể, nhận diện khuôn mặt. Xử lý ngôn ngữ tự nhiên (NLP): Dự đoán từ, dịch máy, phân tích cảm xúc. Hệ thống đề xuất: Dự đoán sở thích người dùng và đề xuất sản phẩm, phim ảnh, nhạc, v.v. Xe tự lái: DNN giúp nhận diện các đối tượng xung quanh và đưa ra quyết định cho các phương tiện tự lái.

#### **1.3.2 Giới thiệu về mô hình RNN**

* **Định nghĩa** : Mô hình Recurrent Neural Network (RNN), hay Mạng Nơ-ron Hồi tiếp, là một loại mạng nơ-ron nhân tạo đặc biệt, được thiết kế để xử lý các dữ liệu có tính chuỗi (sequential data), như văn bản, âm thanh, chuỗi thời gian hoặc bất kỳ dữ liệu nào mà thứ tự các phần tử trong chuỗi có ý nghĩa. Mô hình này có khả năng ghi nhớ thông tin từ các bước thời gian trước (các giá trị đầu vào trước đó), giúp nó có thể xử lý các nhiệm vụ cần phải hiểu và dự đoán theo thời gian.
* **Cấu trúc của RNN** là RNN có một cấu trúc đặc biệt so với các mạng nơ-ron truyền thống. Thay vì chỉ có một lớp đầu vào, lớp ẩn và lớp đầu ra, RNN có các kết nối hồi tiếp giữa các nút trong cùng một lớp ẩn, cho phép mô hình "ghi nhớ" các thông tin từ các bước thời gian trước đó.
* **Hoạt động của RNN** là mô hình RNN hoạt động theo cách mà thông tin không chỉ đi từ lớp đầu vào đến lớp đầu ra, mà còn "quay lại" lớp ẩn trong các bước thời gian sau, giúp mạng có thể ghi nhớ thông tin trước đó. Cụ thể, trong mỗi bước thời gian ttt, RNN tính toán trạng thái ẩn hth\_tht​ theo công thức:

A white background with black text

Description automatically generated

Hình 1(Công thức của RNN)

Sau khi tính toán trạng thái ẩn, mạng sẽ tạo ra đầu ra tại thời điểm ttt, thường được tính theo công thức:

A white background with black text

Description automatically generated

Hình 2(Công thức tính tạo ra đầu ra tại thời điểm t)

* **Ứng dụng của RNN** là RNN rất mạnh mẽ trong các ứng dụng cần phải xử lý dữ liệu theo chuỗi thời gian, nơi các yếu tố trong chuỗi phụ thuộc vào nhau. Xử lý ngôn ngữ tự nhiên (NLP): RNN có thể được sử dụng cho các nhiệm vụ như dịch máy, phân loại văn bản, phân tích cảm xúc, hay tạo văn bản tự động. Nhận dạng giọng nói: RNN có thể được áp dụng để nhận diện và chuyển đổi lời nói thành văn bản, vì chúng có khả năng xử lý chuỗi âm thanh liên tục. Dự đoán chuỗi thời gian: RNN có thể dự đoán các giá trị trong tương lai dựa trên các giá trị lịch sử (ví dụ: dự báo giá cổ phiếu, thời tiết, hay tiêu thụ năng lượng). Sinh dữ liệu âm nhạc hoặc hình ảnh: Một số mô hình RNN có thể sinh ra dữ liệu mới (ví dụ: tạo nhạc, tạo văn bản hoặc hình ảnh).
* **Lợi ích của RNN** là khả năng xử lý dữ liệu chuỗi: RNN đặc biệt mạnh mẽ trong việc xử lý các dữ liệu có tính tuần tự, nơi thứ tự các phần tử trong chuỗi có ý nghĩa quan trọng. Khả năng ghi nhớ thông tin quá khứ: Nhờ vào kết nối hồi tiếp, RNN có thể duy trì trạng thái ẩn qua các bước thời gian, cho phép mô hình "ghi nhớ" thông tin từ quá khứ và sử dụng chúng để dự đoán các bước tiếp theo.
* **Nhược điểm của RNN** là Vấn đề vanishing/exploding gradient: Khi độ sâu của mạng tăng lên (số bước thời gian nhiều), RNN có thể gặp phải vấn đề vanishing gradient (gradient biến mất) hoặc exploding gradient (gradient bùng nổ), khiến quá trình huấn luyện trở nên khó khăn và không hiệu quả. Khó khăn trong việc học các mối quan hệ dài hạn: Mặc dù RNN có thể ghi nhớ thông tin từ các bước thời gian trước đó, nhưng đối với các chuỗi dài, nó khó có thể duy trì thông tin từ quá khứ xa. Điều này dẫn đến việc RNN khó học được các mối quan hệ dài hạn trong chuỗi dữ liệu.

#### **1.3.3 Giới thiệu về LSTM**

* **Định nghĩa :** LSTM (Long Short-Term Memory) là một loại mạng nơ-ron hồi tiếp (Recurrent Neural Network - RNN) đặc biệt, được thiết kế để giải quyết vấn đề vanishing gradient và exploding gradient mà các mạng RNN truyền thống gặp phải khi xử lý các chuỗi dài. LSTM được phát triển bởi Sepp Hochreiter và Jürgen Schmidhuber vào năm 1997, và đã trở thành một công cụ cực kỳ quan trọng trong việc xử lý dữ liệu chuỗi thời gian và các tác vụ như nhận diện giọng nói, xử lý ngôn ngữ tự nhiên và dự đoán chuỗi thời gian.
* **Cấu trúc của LSTM** là khác với các RNN truyền thống ở chỗ nó có một cấu trúc đặc biệt giúp duy trì thông tin trong thời gian dài. Mô hình LSTM bao gồm các cổng (gates), mỗi cổng có nhiệm vụ điều khiển thông tin được lưu trữ hoặc loại bỏ trong tế bào bộ nhớ (memory cell). Cổng quên (Forget Gate): Quyết định thông tin nào từ tế bào bộ nhớ cần được loại bỏ. Nó nhận đầu vào từ trạng thái ẩn của bước trước và đầu vào tại bước hiện tại, sau đó sử dụng một hàm sigmoid để xác định mức độ thông tin cần loại bỏ. Cổng nhập (Input Gate): Xác định lượng thông tin mới nào sẽ được lưu trữ trong tế bào bộ nhớ. Cổng này cũng bao gồm một hàm sigmoid và một hàm tanh, giúp tạo ra các giá trị mới sẽ được ghi vào bộ nhớ. Cổng xuất (Output Gate): Quyết định thông tin nào sẽ được truyền ra ngoài, tức là được sử dụng cho dự đoán tại bước thời gian hiện tại. Nó sử dụng trạng thái bộ nhớ và một hàm sigmoid để quyết định phần tử nào của tế bào bộ nhớ sẽ được xuất ra. Công thức :

A white sheet with black text and black text

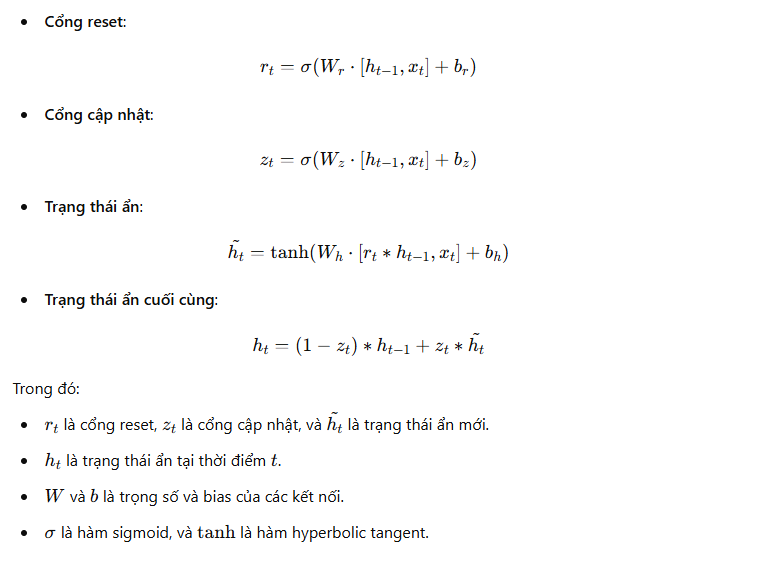
Description automatically generated

Hình 3(Công thức LSTM)

* **Hoạt động của LSTM** là LSTM hoạt động giống như RNN trong việc xử lý chuỗi dữ liệu, nhưng với các cổng và bộ nhớ, nó có thể học và duy trì thông tin trong một khoảng thời gian dài hơn. Cấu trúc này giúp LSTM ghi nhớ thông tin quan trọng từ các bước thời gian trước và bỏ qua những thông tin không cần thiết, giúp nó có khả năng học các mối quan hệ dài hạn trong dữ liệu chuỗi.
* **Ưu điểm của LSTM** là khả năng xử lý mối quan hệ dài hạn: LSTM đặc biệt hữu ích trong việc xử lý các chuỗi dài vì khả năng ghi nhớ thông tin trong thời gian dài mà không bị mất mát thông tin, điều mà các RNN thông thường không thể làm tốt. Giảm thiểu vấn đề vanishing/exploding gradient: LSTM giải quyết vấn đề vanishing gradient (gradient biến mất) bằng cách duy trì một "cấu trúc bộ nhớ" ổn định, giúp quá trình huấn luyện hiệu quả hơn. Khả năng học chuỗi phức tạp: LSTM có thể học được các mẫu phức tạp trong chuỗi dữ liệu, giúp cải thiện hiệu quả trong các bài toán như phân loại văn bản, dự báo chuỗi thời gian, và nhận dạng giọng nói.
* **Ứng dụng của LSTM** là xử lý ngôn ngữ tự nhiên (NLP): LSTM được sử dụng trong các bài toán như dịch máy, phân loại văn bản, phân tích cảm xúc, và trả lời câu hỏi tự động. Nhận diện giọng nói: Trong các hệ thống nhận diện giọng nói, LSTM có thể học các mô hình chuỗi âm thanh phức tạp và giúp cải thiện độ chính xác của nhận diện. Dự báo chuỗi thời gian: LSTM được sử dụng để dự báo giá cổ phiếu, biến động thị trường, hoặc bất kỳ loại dữ liệu chuỗi thời gian nào như dự báo thời tiết hoặc nhu cầu năng lượng. Sinh dữ liệu: LSTM cũng có thể được sử dụng trong các tác vụ tạo ra dữ liệu mới, như sinh văn bản, nhạc, hoặc hình ảnh.
* **Nhược điểm của LSTM** là khó huấn luyện: Dù LSTM giúp giải quyết vấn đề vanishing gradient, nhưng việc huấn luyện mô hình LSTM vẫn có thể gặp khó khăn, đặc biệt khi làm việc với các tập dữ liệu lớn. Độ phức tạp tính toán cao: Mô hình LSTM yêu cầu tính toán phức tạp và nhiều tài nguyên, điều này có thể gây khó khăn khi triển khai trên các thiết bị có tài nguyên hạn chế.

#### **1.3.5 Giới thiệu về GRU**

* **Định nghĩa :** GRU (Gated Recurrent Unit) là một biến thể của mạng nơ-ron hồi tiếp (Recurrent Neural Network - RNN) được phát triển nhằm giải quyết một số vấn đề mà mạng RNN truyền thống gặp phải, chẳng hạn như vanishing gradient (gradient biến mất) và exploding gradient (gradient bùng nổ), đặc biệt khi xử lý chuỗi dữ liệu dài. GRU được đề xuất bởi Kyunghyun Cho và các cộng sự vào năm 2014, như một sự cải tiến của LSTM (Long Short-Term Memory). Mặc dù GRU được thiết kế đơn giản hơn LSTM, nhưng nó vẫn có khả năng học các mối quan hệ dài hạn trong chuỗi dữ liệu. GRU ngày càng được sử dụng rộng rãi nhờ vào khả năng giảm độ phức tạp tính toán và cải thiện hiệu suất trong một số ứng dụng so với LSTM.
* **Cấu trúc của GRU** là GRU có cấu trúc đơn giản hơn LSTM vì chỉ sử dụng hai cổng chính thay vì ba cổng như LSTM. Các cổng trong GRU giúp điều khiển thông tin được lưu trữ, cập nhật hoặc bỏ qua trong quá trình huấn luyện. Cổng cập nhật (Update Gate): Cổng này quyết định bao nhiêu phần của thông tin mới từ bước hiện tại cần được kết hợp với thông tin từ bước trước. Nó giúp mạng chọn lọc thông tin cần thiết và cập nhật trạng thái của tế bào bộ nhớ. Cổng cập nhật có thể coi là sự kết hợp của cả cổng quên và cổng nhập trong LSTM. Cổng reset (Reset Gate): Cổng này quyết định mức độ ảnh hưởng của thông tin quá khứ (trạng thái ẩn của bước trước) đối với bước hiện tại. Cổng reset giúp xác định liệu mạng có nên "quên" thông tin cũ và chỉ sử dụng thông tin mới từ bước hiện tại hay không.
* **Hoạt động của GRU** là mô hình GRU hoạt động tương tự như RNN, nhưng với các cổng điều khiển thông tin mạnh mẽ. Các công thức chính trong GRU bao gồm:



Hình 4 Công thức GRU

* **Ưu điểm của GRU** là Cấu trúc đơn giản hơn LSTM: GRU có chỉ hai cổng (reset và cập nhật) thay vì ba cổng (quên, nhập và xuất) như trong LSTM. Điều này giúp giảm bớt độ phức tạp và giảm số lượng tham số cần huấn luyện, làm cho GRU nhanh hơn và dễ huấn luyện hơn trong một số tình huống. Hiệu suất tương đương hoặc tốt hơn so với LSTM: Mặc dù cấu trúc của GRU đơn giản hơn, nhưng trong nhiều trường hợp, GRU có thể đạt hiệu suất tương đương hoặc thậm chí tốt hơn so với LSTM, đặc biệt khi làm việc với dữ liệu nhỏ hoặc tài nguyên tính toán hạn chế. Dễ dàng triển khai và huấn luyện: Với số lượng tham số ít hơn, GRU yêu cầu ít tài nguyên tính toán và thời gian huấn luyện hơn LSTM. Điều này giúp GRU trở thành sự lựa chọn phổ biến trong các ứng dụng thực tế, đặc biệt khi xử lý các bộ dữ liệu lớn hoặc khi cần triển khai trên các thiết bị với tài nguyên hạn chế.
* **Ứng dụng của GRU** là Xử lý ngôn ngữ tự nhiên (NLP): GRU có thể được sử dụng trong các bài toán như dịch máy, phân loại văn bản, phân tích cảm xúc, và trả lời câu hỏi tự động. Nhận diện giọng nói: GRU có thể học các mô hình chuỗi âm thanh và giúp cải thiện độ chính xác trong nhận diện giọng nói. Dự báo chuỗi thời gian: GRU được áp dụng trong các bài toán dự báo chuỗi thời gian như dự báo giá cổ phiếu, nhu cầu năng lượng, hoặc thay đổi thời tiết. Sinh dữ liệu: GRU cũng có thể được sử dụng trong các tác vụ sinh dữ liệu mới, chẳng hạn như tạo văn bản tự động, tạo nhạc hoặc hình ảnh.
* **Nhược điểm của GRU** làkhó giải thích: Giống như LSTM, GRU vẫn là một mô hình "hộp đen", điều này có nghĩa là việc giải thích lý do mà mô hình đưa ra một quyết định cụ thể có thể gặp khó khăn. Hiệu quả không phải lúc nào cũng vượt trội: Mặc dù GRU có thể hoạt động tốt trong nhiều ứng dụng, nhưng trong một số trường hợp, LSTM vẫn có thể cho kết quả tốt hơn, đặc biệt khi cần xử lý các chuỗi dài hoặc các dữ liệu phức tạp.

# **CHƯƠNG 2 : QUY TRÌNH THỰC HIỆN**

## **II, Các bước thực hiện**

2.1 mô tả bài toán

**BÀI TOÁN: Dự đoán tiêu thụ năng lượng**

* xây dựng mô hình dự đoán tiêu thụ năng lượng cho các khu vực (Zone 1, Zone 2, Zone 3) dựa trên các yếu tố thời tiết và môi trường (nhiệt độ, độ ẩm, tốc độ gió, và bức xạ mặt trời). Các mô hình học máy như hồi quy tuyến tính, hồi quy cây quyết định, hoặc các mô hình phức tạp hơn như Random Forest hoặc XGBoost có thể được sử dụng để dự đoán tiêu thụ năng lượng theo thời gian.

### **2.1 Thu thập dữ liệu**

* Tập dữ liệu trên Kaggle

A screenshot of a computer

Description automatically generated

Hình 5 Dữ liệu trên Kaggle

* Import thư viện cần sử dụng

A white background with black text

Description automatically generated

Hình 6 Các thư viện

2.2 đọc hiểu dữ liệu

A screenshot of a computer

Description automatically generated

* Mô tả dữ liệu
  + DateTime:

+ Mô tả: Cột này chứa thông tin về thời gian ghi nhận các giá trị dữ liệu. Định dạng là "ngày/tháng/năm giờ:phút".

* + Temperature:

+ Mô tả: Nhiệt độ đo được tại thời điểm tương ứng (đơn vị: °C).

* + Humidity:

+ Mô tả: Độ ẩm không khí tại thời điểm đo (đơn vị: %).

* + Wind Speed:

+ Mô tả: Tốc độ gió tại thời điểm đo (đơn vị: m/s).

* + General Diffuse Flows:

+ Mô tả: Dòng bức xạ tán xạ chung (có thể là bức xạ mặt trời, đơn vị: kW/m²).

* + Diffuse Flows:

+ Mô tả: Dòng bức xạ tán xạ (đơn vị có thể là kW/m²).

* + Zone 1 Power Consumption:

+ Mô tả: Mức tiêu thụ năng lượng tại khu vực 1 (đơn vị: kWh).

* + Zone 2 Power Consumption:

+ Mô tả: Mức tiêu thụ năng lượng tại khu vực 2 (đơn vị: kWh).

* + Zone 3 Power Consumption:

+ Mô tả: Mức tiêu thụ năng lượng tại khu vực 3 (đơn vị: kWh).

2.3 tiền sử lý dữ liệu

- nhìn vào lệnh info ta có thể thấy dữ liệu có 52416 bản ghi và 9 thuộc tính với 3 biến phụ thuộc là biến zone1, zone2, zone3 ở đây ta chọn biến zone1 để xây dựng mô hình dự báo với 5 biến độc lập còn lại.

- không có dữ liệu missing , các kiểu dữ liệu có vẻ đúng ta sẽ đi kiểm tra trung lặp, kiểm tra giá trị ngoại lai và chuẩn hóa dữ liệu.

- kiểm tra dữ liệu trung lặp, và đổi tên các cột để dễ sử lý.

A screenshot of a computer

Description automatically generated

* + Sử lý dữ liệu missing

A screenshot of a computer

Description automatically generated

+ Chúng ta sử dụng biểu đồ boxplot để hiển thị các giá trị ngoại lai: có thẻ thấy các giá trị ngoại lai ở các cột nhiet\_do, do\_am, dong\_buc\_xa\_chung,dong\_buc\_xa\_tx và khu2, khu3

+ với các giá trị ngoại lai này ta nên thay thế hoặc giữ lại chúng để chánh bị mất tính chuỗi thời gian của tập dữ liệu, ở đây chúng ta sẽ thay thế chúng.

A screenshot of a computer

Description automatically generated

A white background with colorful text

Description automatically generated

Hàm để thay thế ngoại lai bằng giá trị trên dưới và kiểm tra lại xem còn ngoại lai không

A screenshot of a graph

Description automatically generated

* + Chuẩn hóa dữ liệu

A white rectangular object with colorful text

Description automatically generated

A screenshot of a computer

Description automatically generated

III: xây dựng mô hình

3.1 mô hình dnn

A screenshot of a computer code

Description automatically generated

A screenshot of a computer

Description automatically generated

aA screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screen shot of a computer screen

Description automatically generated

 **Nhận xét từ biểu đồ**:

* Đường dự đoán (Predicted) không theo sát đường thực tế (Actual), đặc biệt là ở các giai đoạn có dao động lớn.
* Điều này là do DNN không được thiết kế để xử lý dữ liệu tuần tự (sequential data), vì vậy nó không "nhớ" được các mối liên hệ dài hạn trong dữ liệu chuỗi thời gian.

 **Ưu điểm**:

* DNN đơn giản, dễ triển khai, và có thể học tốt các mẫu dữ liệu phi tuyến tính trong trường hợp mối quan hệ dữ liệu không phụ thuộc vào thời gian.

 **Hạn chế**:

* DNN thiếu khả năng "ghi nhớ" thông tin quá khứ, dẫn đến việc dự đoán thiếu chính xác khi dữ liệu có sự phụ thuộc theo thời gian.

 **Cần cải thiện**:

* Nếu bạn muốn tiếp tục sử dụng DNN, hãy thử thêm các đặc trưng thời gian (time features) như độ trễ (lag features) hoặc các chỉ báo chuỗi (e.g., trung bình động).
* Tuy nhiên, tốt hơn là sử dụng các mô hình chuyên xử lý chuỗi như RNN, LSTM, hoặc GRU.

3.2 mô hình rnn

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screen shot of a computer screen

Description automatically generated

 **Nhận xét từ biểu đồ**:

* RNN đã cải thiện đáng kể so với DNN, với đường dự đoán (Predicted) có xu hướng gần hơn với đường thực tế (Actual).
* Tuy nhiên, sai lệch vẫn còn khá lớn, đặc biệt ở các điểm dao động lớn (biên độ cao).

 **Ưu điểm**:

* RNN có khả năng xử lý dữ liệu tuần tự tốt hơn DNN, vì nó có thể ghi nhớ thông tin từ các bước thời gian trước đó.
* Biểu đồ cho thấy mô hình nắm bắt được xu hướng tổng thể.

 **Hạn chế**:

* Gặp vấn đề vanishing gradient, khiến RNN khó học được các mối quan hệ dài hạn trong chuỗi thời gian.
* Sai số lớn ở các điểm biến động cao.

 **Cần cải thiện**:

* Sử dụng LSTM hoặc GRU để cải thiện khả năng dự đoán, đặc biệt với chuỗi dữ liệu dài và biến động lớn.
* Điều chỉnh các tham số như số bước thời gian (time steps) hoặc số lượng đơn vị ẩn (hidden units).

3.3 mô hình ltsm

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generatedA screen shot of a graph

Description automatically generated

* **Nhận xét từ biểu đồ**:
  + LSTM dự đoán sát hơn với giá trị thực tế so với RNN, đặc biệt ở các điểm có dao động lớn.
  + Đường dự đoán (Predicted) đã khớp với xu hướng của đường thực tế (Actual) tốt hơn, mặc dù vẫn còn sai lệch nhỏ.
* **Ưu điểm**:
  + LSTM giải quyết vấn đề vanishing gradient và có khả năng ghi nhớ mối quan hệ dài hạn trong chuỗi.
  + Hiệu quả cao hơn RNN trong việc xử lý các chuỗi dài và phức tạp.
* **Hạn chế**:
  + Thời gian huấn luyện lâu hơn và yêu cầu tài nguyên tính toán lớn.
  + Sai lệch nhỏ vẫn còn ở các điểm dao động mạnh.
* **Cần cải thiện**:
  + Tối ưu hóa siêu tham số như learning rate, batch size, và số lớp ẩn.
  + Tăng cường dữ liệu đầu vào (data augmentation) hoặc loại bỏ nhiễu trong dữ liệu.

3.4 mô hình gru

A screenshot of a computer program

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generated

A screen shot of a computer screen

Description automatically generated

* **Nhận xét từ biểu đồ**:
  + GRU cho kết quả tương đương hoặc thậm chí tốt hơn LSTM ở một số giai đoạn.
  + Đường dự đoán (Predicted) khớp rất tốt với đường thực tế (Actual), đặc biệt ở các giai đoạn dao động vừa phải.
* **Ưu điểm**:
  + GRU đơn giản hơn LSTM, với số lượng tham số ít hơn nhưng vẫn giữ được khả năng xử lý mối quan hệ dài hạn.
  + Dự đoán tốt hơn so với RNN và tương đương LSTM, với thời gian huấn luyện ngắn hơn.
* **Hạn chế**:
  + Có thể gặp khó khăn trong việc xử lý các chuỗi rất dài hoặc dữ liệu cực kỳ phức tạp.
* **Cần cải thiện**:
  + So sánh thêm với LSTM trên các chỉ số đánh giá như MAE, RMSE để chọn mô hình tối ưu.
  + Tối ưu hóa số lượng đơn vị ẩn và cổng (gates) trong GRU.

**Tổng Kết và Đề Xuất**

* **DNN**: Không phù hợp lắm cho bài toán chuỗi thời gian. Chỉ nên dùng nếu không thể sử dụng các mô hình chuyên xử lý chuỗi.
* **RNN**: Phù hợp ở mức cơ bản nhưng hiệu quả không cao bằng LSTM hoặc GRU. Có thể thay thế bằng LSTM hoặc GRU.
* **LSTM**: Là lựa chọn tốt cho bài toán này, đặc biệt khi chuỗi dữ liệu dài và có các mối quan hệ phức tạp.
* **GRU**: Đơn giản hơn nhưng vẫn hiệu quả như LSTM. Nếu hiệu suất (tốc độ) là yếu tố quan trọng, GRU là lựa chọn tối ưu.