

**TRƯỜNG ĐẠI HỌC THỦY LỢI
KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP LỚN
MÔN : Phân Tích Chuỗi Thời Gian**

**ĐỀ TÀI:
DỰ BÁO BỨC XẠ MẶT TRỜI TỪ DỮ LIỆU BỨC XẠ VÀ ẢNH SKY CAM**

Giảng viên hướng dẫn: Ths. Trần Anh Đạt

Nhóm sinh viên thực hiện: Nhóm 3_64TTNT1

Họ tên sinh viên	Mã sinh viên
Thái Văn Sáng	2251262634
Vũ Thị Như Quỳnh	2251262633
Nguyễn Thị Quỳnh Anh	2251262657
Đông Anh Quân	2251262626

MỤC LỤC

Chương I: Giới thiệu	2
1.1 Bối cảnh và lý do chọn đề tài	2
1.2 Mục tiêu đề tài	2
1.3 Phạm vi thực hiện	2
Chương II: Cơ sở lý thuyết	3
2.1 Khái niệm bức xạ mặt trời và yếu tố ảnh hưởng	3
2.2 Giới thiệu về các mô hình học sâu	3
2.3 Giới thiệu về Pipeline chung của bài	11
2.4 Các chỉ số đánh giá mô hình	11
Chương III: Dữ liệu và tiền xử lý	13
3.1 Tập dữ liệu	13
3.2 Tiền xử lý dữ liệu	15
Chương IV: Mô hình và phương pháp	16
4.1 Mô hình Resnet kết hợp với LSTM	16
4.2 Mô hình Resnet kết hợp với Transformer	17
4.3 Mô hình Resnet kết hợp với Timesformer	18
Chương V: Kết quả và đánh giá	19
5.1 Cấu hình huấn luyện và các độ đo đánh giá	19
5.2 Kết quả thực nghiệm	19
5.3 So sánh giữa các mô hình	22
Chương VI: Kết luận và hướng phát triển	23
6.1 Kết luận	23
6.2 Hướng phát triển	23
TÀI LIỆU THAM KHẢO	24

Chương I. Giới thiệu

1.1 Bối cảnh và lý do chọn đề tài

Bức xạ mặt trời là một trong những nguồn năng lượng tái tạo quan trọng trong bối cảnh chuyển dịch năng lượng toàn cầu [1]. Việc dự báo chính xác bức xạ mặt trời đóng vai trò thiết yếu trong việc điều độ hệ thống điện mặt trời, tối ưu hóa công suất phát và giảm thiểu rủi ro cho lưới điện.

Ảnh Sky Cam – ảnh chụp bầu trời thời gian thực – cung cấp thông tin trực quan về trạng thái mây, là yếu tố ảnh hưởng lớn đến bức xạ. Do đó, việc kết hợp dữ liệu đo bức xạ và ảnh Sky Cam để xây dựng mô hình dự báo thông minh là một hướng tiếp cận mới và đầy hứa hẹn.

1.2 Mục tiêu đề tài

- Xây dựng mô hình dự báo bức xạ mặt trời trong ngắn hạn.
- Kết hợp dữ liệu định lượng (chuỗi thời gian bức xạ) và dữ liệu hình ảnh (Sky Cam).
- Ứng dụng các kỹ thuật học máy và học sâu (CNN, LSTM, Transformer).

1.3 Phạm vi thực hiện

- Thời gian dự báo: từ 5 phút đến 30 phút.
- Dữ liệu thực nghiệm: tập dữ liệu gồm ảnh Sky Cam và đo đạc bức xạ tại một trạm cụ thể
- Mô hình thử nghiệm: thử nghiệm các loại mô hình kết hợp như: Resnet+LSTM, Resnet+Transformer, Resnet+TimesFormer

Chương II. Cơ sở lý thuyết

2.1 Khái niệm bức xạ mặt trời và yếu tố ảnh hưởng

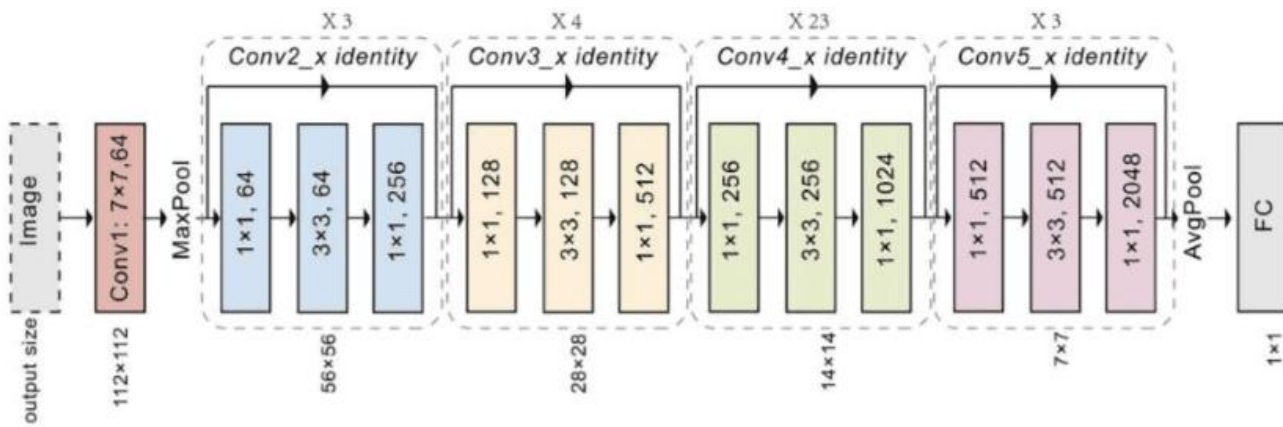
Bức xạ mặt trời là năng lượng mà Mặt Trời phát ra dưới dạng sóng điện từ. Năng lượng này di chuyển qua không gian và đến Trái Đất, đóng vai trò cực kỳ quan trọng trong việc duy trì sự sống và điều khiển các quá trình khí hậu – thời tiết trên hành tinh của chúng ta. Bức xạ mặt trời chính là ánh sáng và nhiệt từ Mặt Trời, nó bao gồm nhiều loại tia nhưng chủ yếu là: Ánh sáng (mắt người có thể nhìn thấy), Tia hồng ngoại (IR: mang nhiệt), Tia tử ngoại (UV: không nhìn thấy được, có thể gây cháy nắng thậm chí là ung thư da). Tổng năng lượng từ mặt trời chiếu xuống Trái đất, thường được chia thành 3 thành phần: Global Horizontal Irradiance (GHI), Direct Normal Irradiance (DNI) và Diffuse Horizontal Irradiance (DHI). Trong đó, GHI là thành phần quan trọng nhất được sử dụng trong các hệ thống điện mặt trời.

Một trong những yếu tố ảnh hưởng lớn đến bức xạ là trạng thái mây trong khí quyển. Mây có thể làm giảm mạnh hoặc gây dao động bất thường trong cường độ bức xạ. Các đặc trưng như độ dày mây, mức độ che phủ, chuyển động của mây đều ảnh hưởng trực tiếp đến khả năng hấp thụ bức xạ tại mặt đất. Ảnh Sky Cam là một công cụ hiệu quả để quan sát và trích xuất các thông tin liên quan đến trạng thái mây nhằm phục vụ cho việc dự báo chính xác hơn.

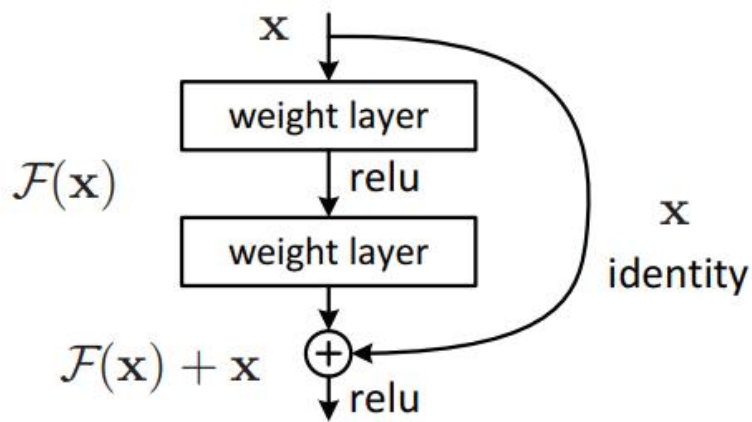
2.2 Giới thiệu về các mô hình học sâu

2.2.1 Giới thiệu về ResNet

- Mô hình ResNet (Residual Network) là một trong những kiến trúc mạng nơ-ron tích chập (CNN) nổi bật, được giới thiệu vào năm 2015 bởi Kaiming He và các cộng sự.
- Cấu trúc: Cấu trúc của ResNet được xây dựng dựa trên việc chia nhỏ mạng thành các khối residual (Residual Blocks), trong đó mỗi khối bao gồm một số lớp tích chập (convolutional layers) cùng với một kết nối tắt trực tiếp từ đầu vào đến đầu ra của khối. Ý tưởng chính là việc học một hàm residual $F(x)$ thay vì học trực tiếp một hàm mục tiêu $H(x)$. Điều này giúp duy trì tính ổn định và độ chính xác khi mạng có nhiều tầng.



Hình 2.2.1.1: Cấu trúc của ResNet [3]



Hình 2.2.1.2: Khối residual cơ bản [3]

- Một khối residual cơ bản trong ResNet có cấu trúc như sau:

$$y = F(x, \{W_i\}) + x$$

Trong đó:

+ x là đầu vào của khối.

+ y là đầu ra của khối.

+ $F(x, \{W_i\})$ là hàm residual, thường được biểu diễn bằng một chuỗi các lớp tích chập với trọng số W_i .

- Nguyên lý hoạt động: Nguyên lý hoạt động của ResNet dựa trên việc thay vì học trực tiếp hàm $H(x)$, mô hình sẽ học một hàm residual $F(x) = H(x) - x$. Sau đó, đầu ra của mô hình sẽ là: $H(x) = F(x) + x$

+ Cách tiếp cận này dựa trên giả định rằng việc học một hàm sai khác $F(x)$ sẽ dễ dàng

hơn so với việc học trực tiếp hàm $H(x)$. Khi mô hình học được $F(x)$, nó thực chất đang học cách “điều chỉnh” đầu vào x để đạt được đầu ra mong muốn $H(x)$. Nhờ vào các kết nối tắt (skip connections), các tín hiệu thông tin có thể dễ dàng được truyền qua mạng mà không bị suy giảm hoặc mất mát, đặc biệt khi mạng trở nên rất sâu.

+ Các kết nối tắt trong ResNet mang lại nhiều lợi ích, bao gồm:

Giảm thiểu vanishing gradient: Khi mạng trở nên rất sâu, các giá trị gradient có thể bị giảm mạnh (vanishing) hoặc tăng mạnh (exploding), gây khó khăn trong quá trình huấn luyện. Kết nối tắt giúp giữ lại các giá trị gradient, giảm thiểu hiện tượng này.

Dễ dàng học các đặc trưng: Việc học một hàm residual $F(x)$ thường đơn giản hơn so với việc học một hàm ánh xạ hoàn toàn mới $H(x)$, do đó mô hình có thể nhanh chóng đạt được độ chính xác cao hơn.

Tăng độ sâu mà không giảm hiệu suất: Với các kết nối tắt, ResNet có thể được mở rộng đến hàng trăm hoặc hàng ngàn tầng mà không gặp phải vấn đề suy giảm hiệu suất.

- **ResNet-50:** ResNet-50 bao gồm 50 tầng, với cấu trúc cụ thể như sau:

+ Conv1: Tầng tích chập đầu tiên với kernel size 7×7 , stride 2, và 64 filters.

+ MaxPool: Lớp MaxPooling với kernel size 3×3 và stride 2.

+ Conv2_x: Gồm 3 khối residual với 64 filters mỗi khối.

+ Conv3_x: Gồm 4 khối residual với 128 filters mỗi khối.

+ Conv4_x: Gồm 6 khối residual với 256 filters mỗi khối.

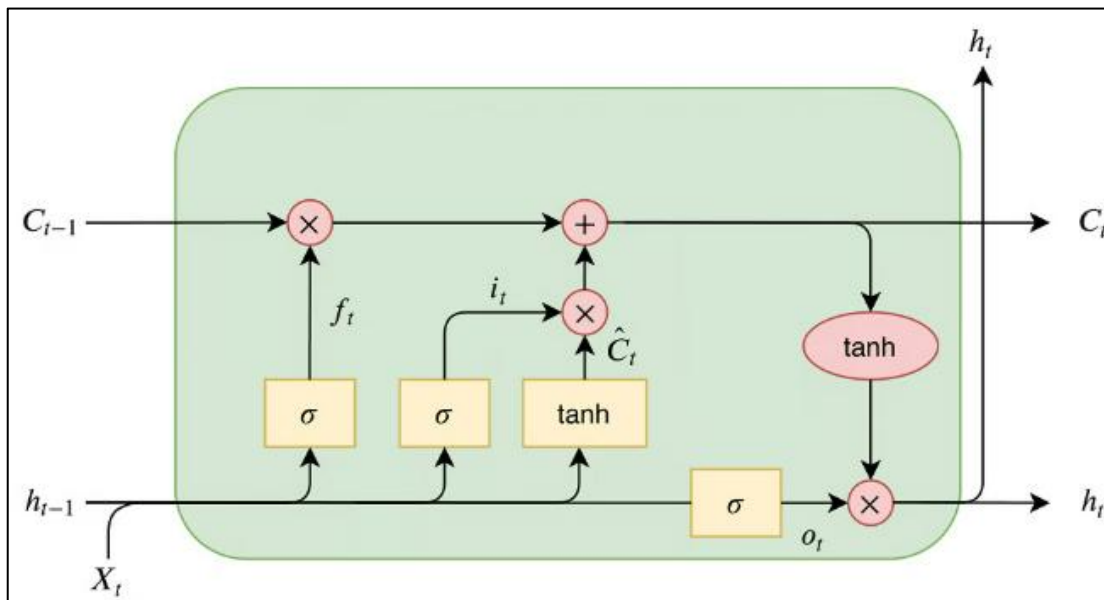
+ Conv5_x: Gồm 3 khối residual với 512 filters mỗi khối.

+ Average Pooling: Lớp pooling trung bình.

+ Fully Connected Layer: Lớp kết nối đầy đủ với số lượng neurons bằng số lớp đầu ra (output classes).

Mỗi khối residual trong các tầng này có cấu trúc tương tự nhau, với một kết nối tắt từ đầu vào đến đầu ra của khối.

2.2.2 Giới thiệu về LSTM



Hình 2.2.2.1 : Tổng quan kiến trúc mô hình LSTM

- LSTM (Long Short-Term Memory) là một loại mạng nơ-ron hồi tiếp (RNN) được thiết kế để xử lý dữ liệu chuỗi thời gian hoặc dữ liệu có tính tuần tự như: dự báo thời tiết, giá cổ phiếu, hoặc trong trường hợp này là dự báo bức xạ mặt trời.

- Cách hoạt động:

+ Cổng quên (Forget Gate): Quyết định thông tin nào cần loại bỏ khỏi trạng thái bộ nhớ.

+ Cổng đầu vào (Input Gate): Quyết định thông tin nào sẽ được thêm vào bộ nhớ.

+ Cổng đầu ra (Output Gate): Xác định thông tin nào sẽ được đưa ra làm đầu ra tại mỗi bước thời gian.

→ Nhờ cơ chế này, LSTM ghi nhớ được cả thông tin ngắn hạn và dài hạn, giúp dự báo chuỗi thời gian chính xác hơn.

- Ứng dụng trong đề tài: Trong đề tài này, LSTM được dùng để học quy luật biến thiên của ảnh và bức xạ mặt trời theo thời gian, từ đó dự báo giá trị trong tương lai dựa trên dữ liệu trong quá khứ.

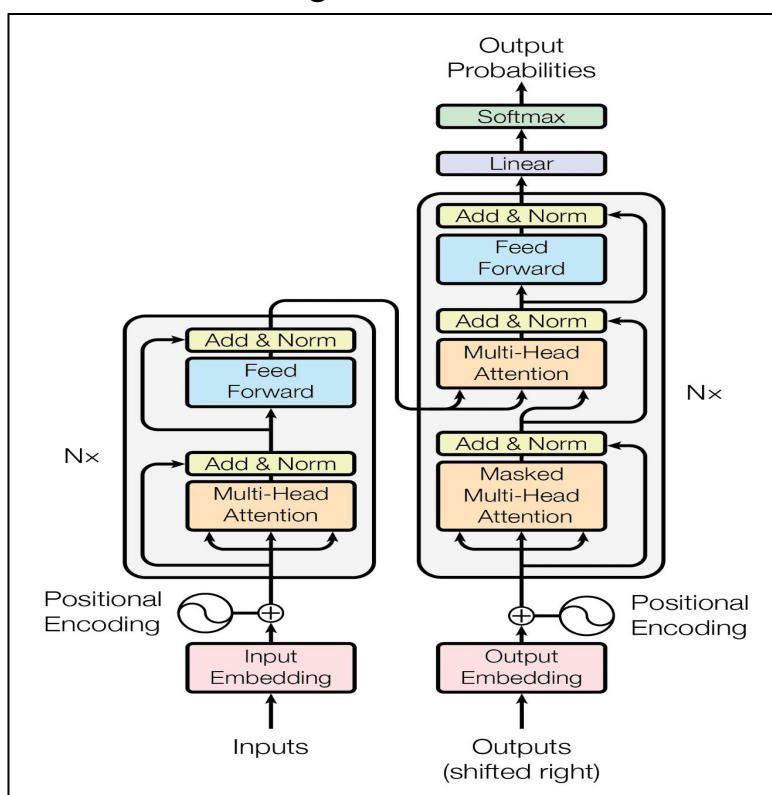
2.2.3 Giới thiệu về Transformer

- Transformer là một **kiến trúc mạng học sâu (deep learning architecture)** được giới thiệu vào năm 2017 trong bài báo nổi tiếng “*Attention is All You Need*” của Vaswani và cộng sự. Mô hình này được thiết kế để xử lý dữ liệu dạng chuỗi (sequences), đặc biệt là trong xử lý ngôn ngữ tự nhiên (NLP), mà không cần sử dụng các mạng tuần tự truyền thống như RNN hay LSTM.

- Cấu trúc tổng thể của Transformer gồm 2 phần chính:

+ **Encoder:** Mã hóa chuỗi đầu vào thành vector đặc trưng.

+ **Decoder:** Giải mã vector đặc trưng để sinh chuỗi đầu ra.



Hình 2.2.3: Tổng quát cấu trúc Transformer

A. Encoder

- Encoder gồm **N lớp (layer)** xếp chồng, mỗi lớp có cấu trúc giống nhau và bao gồm các thành phần:

a. Multi-Head Self-Attention

$$\text{Self-Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) V$$

- + Các token trong chuỗi đầu vào "nhìn" lẫn nhau để học được mối quan hệ giữa các từ, bất kể khoảng cách xa hay gần.
- + Input là embedding của chuỗi đầu vào.
- + Output là trọng số attention và vector đặc trưng đã được tổng hợp.

b. Feed-Forward Network (FFN)

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

- + Một mạng perceptron áp dụng độc lập cho từng token sau khi qua attention.
- + Thường gồm 2 lớp fully-connected kèm hàm kích hoạt phi tuyến ReLU hoặc GELU.

c. Add & Norm (Residual + Layer Normalization)

$$\text{output} = \text{LayerNorm}(x + \text{Sublayer}(x))$$

- + Áp dụng skip-connection và chuẩn hóa để ổn định quá trình huấn luyện, tránh mất thông tin.

B. Decoder

- Decoder cũng gồm **N lớp (layer)** xếp chồng với cấu trúc tương tự nhau, nhưng mỗi lớp gồm 3 thành phần chính:

a. Masked Multi-Head Self-Attention

$$\text{MaskedAttention}(Q, K, V) = \text{softmax}\left(\frac{QK^T + \text{mask}}{\sqrt{d_k}}\right)V$$

- + Các token trong chuỗi đầu ra tự chú ý lẫn nhau.
- + Sử dụng mặt nạ (mask) để đảm bảo mỗi token chỉ tham chiếu các token trước đó, tránh nhìn "tương lai" (autoregressive decoding).

b. Multi-Head Cross-Attention

$$\text{CrossAttention}(Q_{dec}, K_{enc}, V_{enc})$$

- + Các token đầu ra tham chiếu vào vector đặc trưng của Encoder để lấy thông tin từ chuỗi đầu vào.

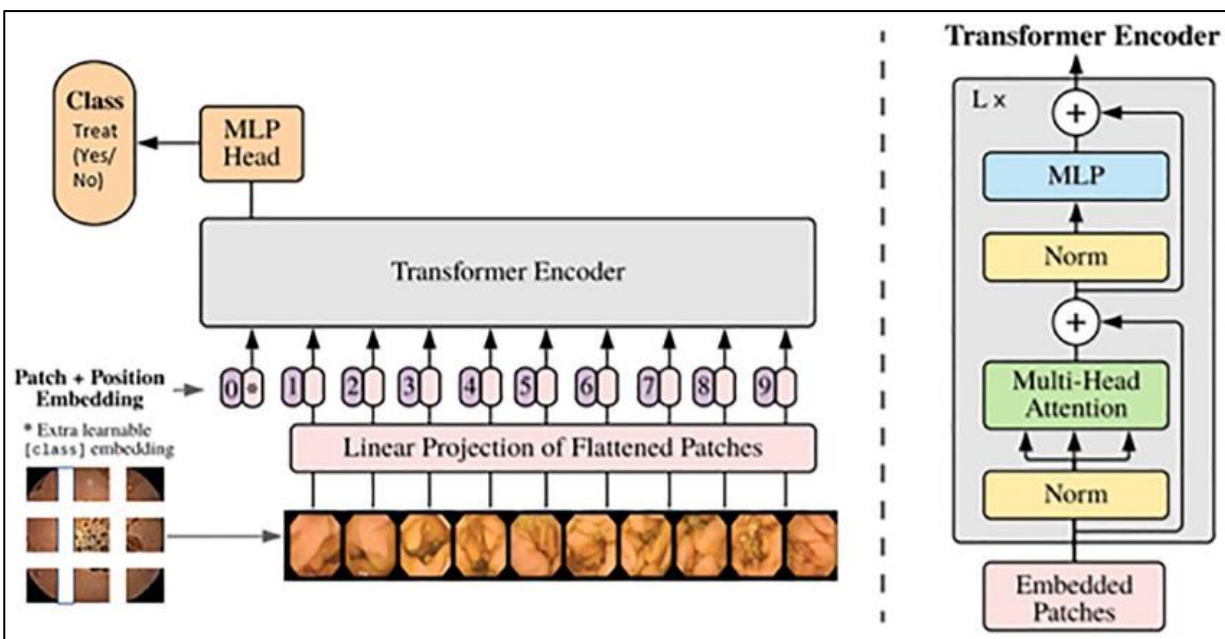
c. Feed-Forward Network (FFN)

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

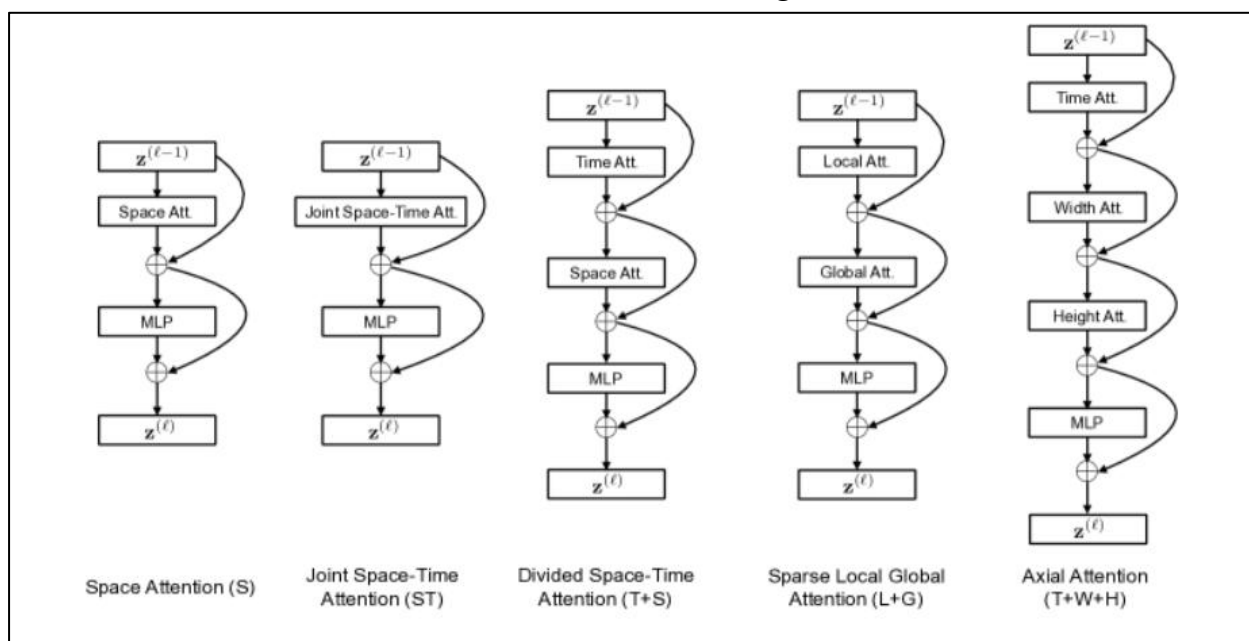
- + Mạng perceptron fully-connected áp dụng độc lập cho từng token, tương tự như trong Encoder.
- Giống Encoder, giữa các thành phần đều có skip-connection và LayerNorm để tối ưu quá trình huấn luyện và truyền gradient.

2.2.4 Giới thiệu về Timesformer

- Khái niệm: **TimeSformer (Time-Space Transformer)** là một kiến trúc Transformer mở rộng từ Vision Transformer (ViT), được thiết kế để xử lý dữ liệu **dạng chuỗi hình ảnh** (video hoặc ảnh chụp có tính chu kỳ). Mô hình áp dụng cơ chế **self-attention tách biệt theo chiều thời gian và không gian**, giúp học được biểu diễn không gian–thời gian hiệu quả.
- Kiến trúc tổng thể



Hình 2.2.4.1: Kiến trúc tổng thể



Hình 2.2.4.2: Cấu trúc khối attention trong TimeSformer

- Dữ liệu đầu vào:
 - + Video: chuỗi khung hình liên tục
 - + Chuỗi ảnh định kỳ: như ảnh bầu trời (10s/lần)
- Dữ liệu được biểu diễn dưới dạng tensor 4 chiều
$$X \in R^{T*H*W*C}$$

Trong đó:

 - + T: số khung hình hoặc số ảnh
 - + H,W: kích thước
 - + C: số kênh
- Patch Embedding
 - + Mỗi ảnh được chia thành các patch nhỏ (ví dụ 16x16 pixel)
 - + Các patch này được ánh xạ qua linear layer thành vector đặc trưng
- Kết quả: là một tensor đầu ra có kích thước:
$$X' \in R^{T \times N \times D}$$

Trong đó:

 - + N: số patch mỗi ảnh
 - + D: kích thước vector đặc trưng
- Positional Encoding
 - + Positional encoding không gian: mã hoá vị trí của từng patch trong ảnh
 - + Positional encoding thời gian: mã hoá thứ tự ảnh trong chuỗi
- Divided Space-Time Attention:
 - + Temporal Attention: Attention giữa cùng một patch qua các thời điểm khác nhau
 - + Học mối quan hệ thời gian
 - Spatial Attention:
 - + Attention giữa các patch trong cùng một ảnh
 - + Học mối quan hệ không gian
- Transformer Encoder Block: Mỗi block Transformer gồm:

LayerNorm → Temporal Attention → LayerNorm Spatial Attention →
MLP Feed-Forward → Residual connection
- Prediction Head: Sau khi dữ liệu đi qua nhiều block attention:
 - + Áp dụng Global Average Pooling để tổng hợp thông tin toàn bộ chuỗi.

+ Đưa qua Linear Layer để thực hiện: Phân loại hành động (nếu bài toán classification). Sau đó, dự đoán giá trị liên tục (nếu bài toán regression, ví dụ dự đoán bức xạ mặt trời).

2.3 Giới thiệu về Pipeline chung của bài

Cách làm của mô hình dựa trên ý tưởng của bài [Paper](#) từ đó nhóm xây dựng Pipeline chung của mô hình đề xuất bao gồm 3 thành phần chính:

- + Trích xuất đặc trưng ảnh: Sử dụng mô hình Resnet để trích xuất đặc trưng từ ảnh bầu trời.
- + Xử lý chuỗi thời gian bằng các mô hình LSTM hoặc Transformer : Đặc trưng của chuỗi ảnh và chuỗi dữ liệu bức xạ tương ứng được đưa vào mô hình để học được mối quan hệ biến động theo thời gian.
- + Kết hợp với dữ liệu bức xạ và thời gian: Chuỗi vector đặc trưng về dữ liệu ảnh và bức xạ được Concat lại với nhau, sau đó thông tin được chuyển qua một tầng fully connected để đưa ra dự đoán.

2.4 Các chỉ số đánh giá mô hình

2.4.1. MAE (Mean Absolute Error)

- **MAE** là chỉ số đo lường sai số trung bình tuyệt đối giữa giá trị dự báo và giá trị thực tế. Nó được tính bằng cách lấy trung bình cộng của giá trị tuyệt đối của sai số tại từng điểm:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}|$$

- Trong đó:
 - + y_i là giá trị thực tế
 - + \hat{y} là giá trị dự đoán
 - + N là số lượng mẫu

2.4.3. MSE (Mean Square Error)

- **MSE** là chỉ số đo lường sai số bình phương trung bình giữa giá trị dự báo và giá trị thực tế. Nó được tính bằng cách lấy trung bình của bình phương sai số tại mỗi điểm:

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y})^2$$

- Trong đó:
- + y_i là giá trị thực tế
- + \hat{y} là giá trị dự đoán
- + N là số lượng mẫu

2.4.2. RMSE (Root Mean Squared Error)

- **RMSE** là chỉ số đo độ lệch trung bình bình phương căn bậc hai giữa giá trị dự báo và giá trị thực tế. Nó phản ánh mức độ sai lệch tiêu chuẩn của dự báo so với thực tế:

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y})^2}$$

- Trong đó:
- + y_i là giá trị thực tế
- + \hat{y} là giá trị dự đoán
- + N là số lượng mẫu

2.4.3. Hệ số xác định R^2 (Coefficient of Determination)

- R^2 là chỉ số cho biết mức độ mà mô hình dự báo giải thích được phương sai (biến động) của dữ liệu thực tế. Nó phản ánh độ phù hợp của mô hình, Mô hình có R^2 càng gần 1 càng tốt:

$$R^2 = 1 - \frac{\sum (y_i - \hat{y})^2}{\sum (y_i - \bar{y})^2}$$

- Trong đó:
- + y_i là giá trị thực tế
- + \hat{y} là giá trị dự đoán
- + \bar{y} : trung bình của giá trị thực tế

Chương III. Dữ liệu và tiền xử lý dữ liệu

3.1 Tập dữ liệu

- Trong dự án này, nhóm sử dụng bộ dữ liệu mang tên "Sky Images and Solar Radiation Measurement Dataset [2]", được cung cấp trên nền tảng Kaggle.

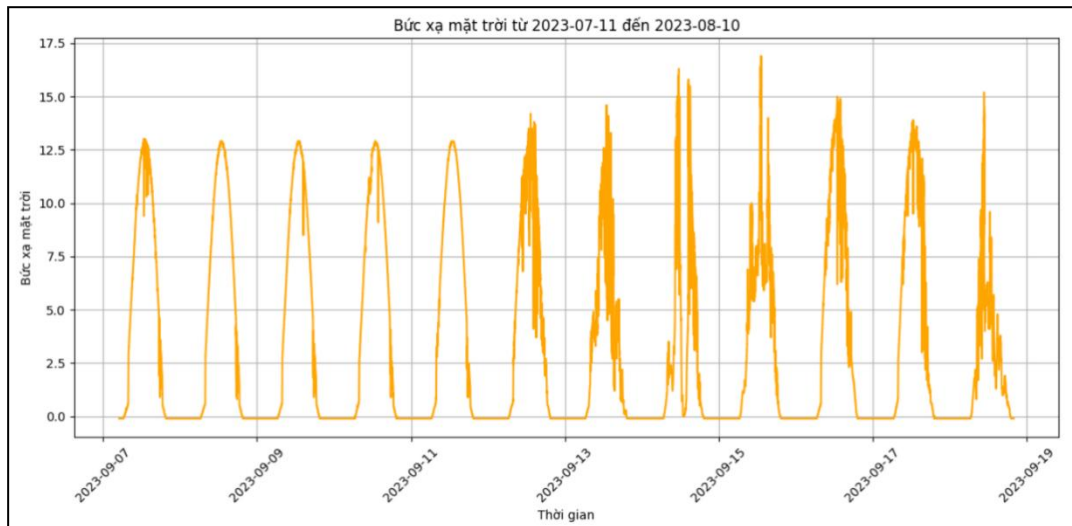
	datetime	radiation	filename
785	2023-09-07 05:00:06	-0.1	/kaggle/input/sky-images-and-solar-radiation-m...
4723	2023-09-07 05:00:16	-0.1	/kaggle/input/sky-images-and-solar-radiation-m...
5662	2023-09-07 05:00:26	-0.1	/kaggle/input/sky-images-and-solar-radiation-m...
1268	2023-09-07 05:00:36	-0.1	/kaggle/input/sky-images-and-solar-radiation-m...
1202	2023-09-07 05:00:46	-0.1	/kaggle/input/sky-images-and-solar-radiation-m...

Hình 3.1.1 Tập dữ liệu

- Bộ dữ liệu này cung cấp hình ảnh bầu trời và thông tin bức xạ mặt trời liên quan được thu thập từ 7/9-18/9/2023 (10.5GB)
- Khoảng thời gian giữa các lần chụp ảnh là 10 giây và các bức ảnh được chụp từ 5:00 đến 21:00.
- Thời gian và ngày chụp ảnh và thông tin bức xạ mặt trời có thể được tìm thấy trong tên tệp hình ảnh.
- Tên tệp hình ảnh có định dạng: YYYY-MM-DD-HH-MM-SS_SolarRadiationValue



Hình 3.1.2 Ảnh ví dụ về dữ liệu ảnh bức xạ tên: 2023-09-07-06-26-09_0.0.JPG



Hình 3.1.3 Biểu đồ phân bố bức xạ mặt trời từ 7/9/2023-18/9/2023

- Nhận xét:

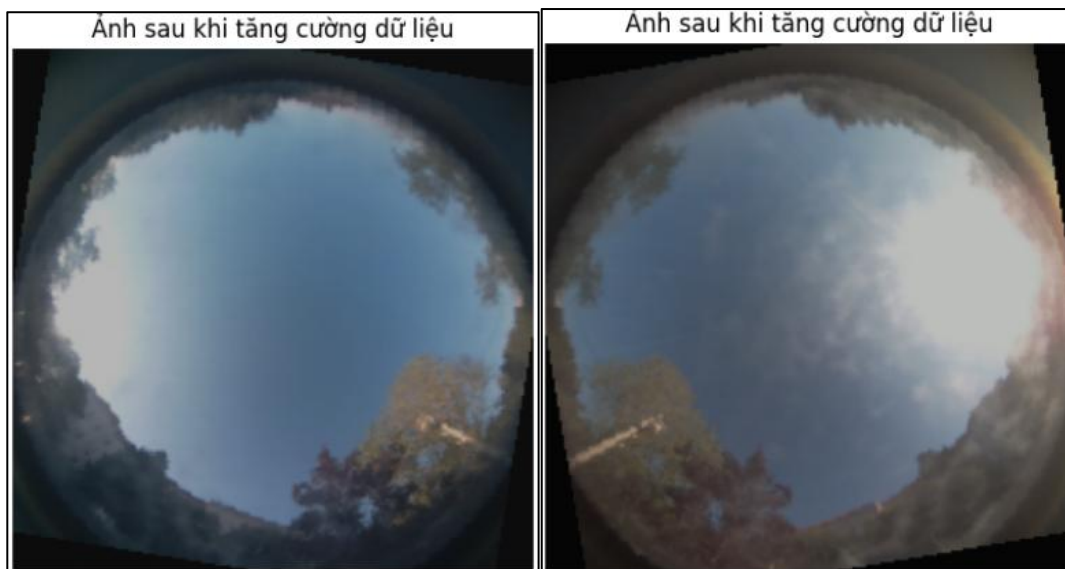
- + Chu kỳ tăng giảm theo ngày rất đều, tăng vào buổi sáng, đạt đỉnh giữa trưa và giảm vào buổi chiều
- + Từ ngày 13/9-18/9 có biến động mạnh hơn (có thể do thời tiết thay đổi, xuất hiện mây, sương mù,...) không mượt và đối xứng như những ngày đầu
- + Mức bức xạ đỉnh dao động khoảng 12-17, và cao nhất trong những ngày có dao động thời tiết rõ rệt
- + Bức xạ giảm về gần 0 vào sáng sớm và chiều muộn đúng với logic khi mặt trời chưa mọc và đã lặn

→ ***Dữ liệu có tính chu kì rất rõ ràng nên tận dụng các mô hình học sâu như LSTM, Transformer với thông tin thời gian***

3.2 Tiền xử lý dữ liệu

3.2.1. Ảnh Sky Cam

- Phương pháp xử lý bao gồm:
 - + Resize ảnh về kích thước 224x224.
 - + Chuẩn hóa các kênh màu theo ImageNet.
 - + Sử dụng các phương pháp tăng cường dữ liệu như: Xoay ảnh, Lật ảnh, Tăng cường độ sáng



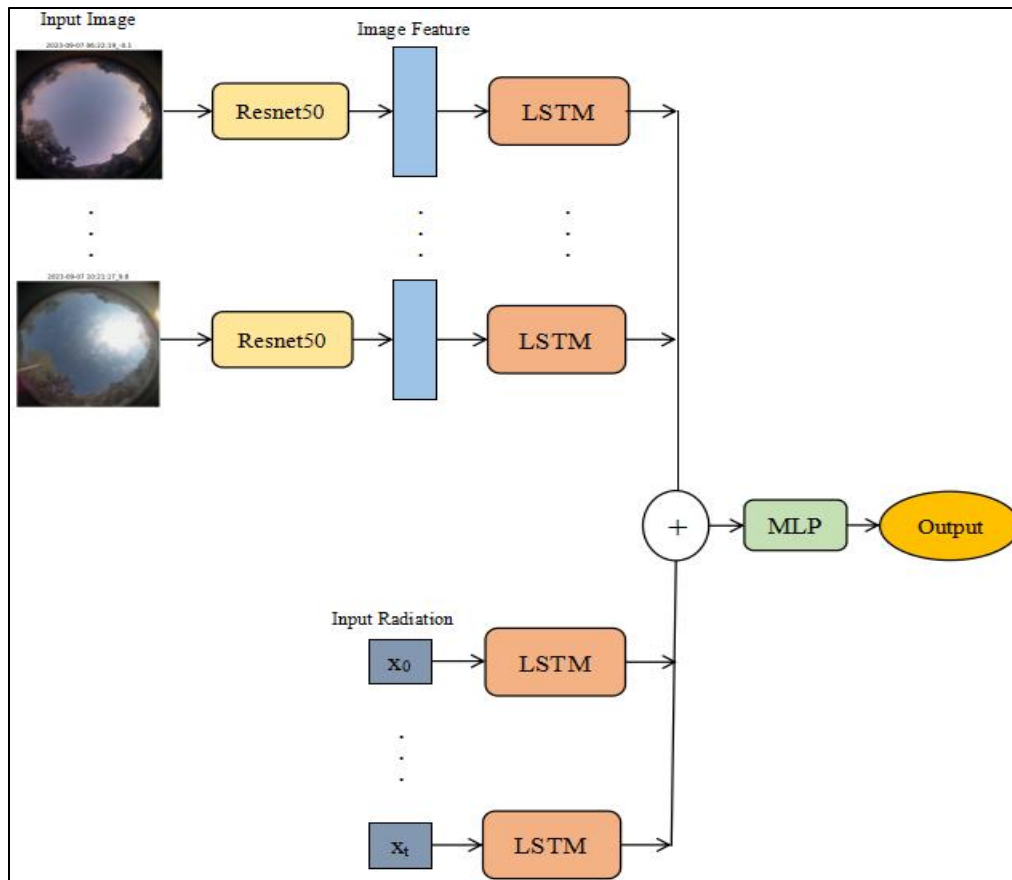
Hình 3.2.1: Ảnh sau khi áp dụng các phương pháp tăng cường dữ liệu

3.2.2. Dữ liệu bức xạ

Do dữ liệu bức xạ có phân bố trong khoảng nhất định và là dữ liệu thực tế được thu thập nên không cần thiết đến việc chuẩn hóa hay xử lý ngoại lai.

Chương IV. Mô hình và phương pháp

4.1 Mô hình Resnet kết hợp với LSTM



Hình 4.1: Mô hình Resnet+LSTM

- Cách hoạt động của mô hình:

Step 1: Mỗi ảnh đầu vào được đưa qua một mạng ResNet50 để trích xuất các đặc trưng hình ảnh.

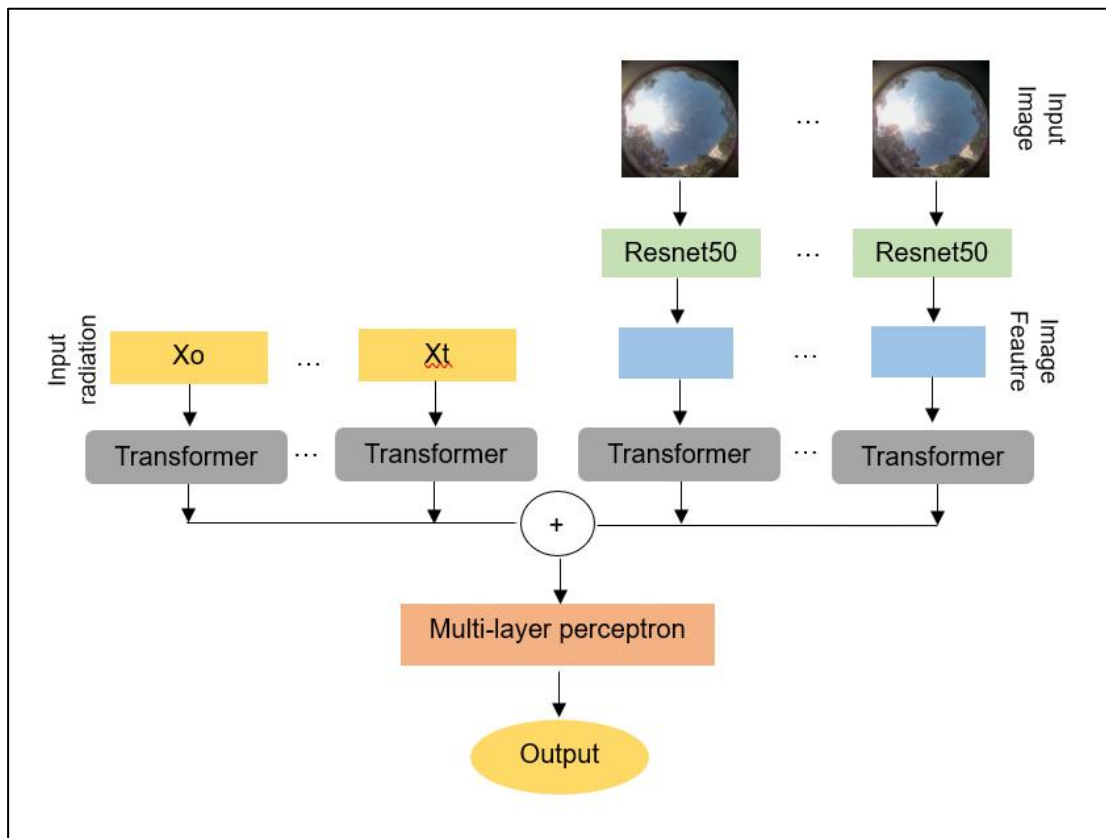
Step 2: Các đặc trưng hình ảnh sau đó được đưa vào một mạng LSTM để xử lý dữ liệu chuỗi các đặc trưng ảnh theo thời gian

Step 3: Dữ liệu bức xạ đầu vào cũng được đưa vào một mạng LSTM riêng biệt để mô hình học và nắm bắt các mẫu hoặc xu hướng trong dữ liệu bức xạ theo thời gian.

Step 4: Đầu ra từ cả hai nhánh LSTM (từ ảnh và từ bức xạ) được Concat lại để kết hợp thông tin từ cả hai nguồn dữ liệu.

Step 5: Kết quả tổng hợp này sau đó được đưa vào một mạng MLP để xử lý các đặc trưng kết hợp này sau đó đưa ra đầu ra cuối cùng của mô hình.

4.2 Mô hình Resnet kết hợp với Transformer



Hình 4.2: Mô hình Resnet+Transformer

- Cách hoạt động của mô hình:

Step 1: Mỗi ảnh đầu vào được đưa qua một mạng ResNet50 để trích xuất các đặc trưng hình ảnh.

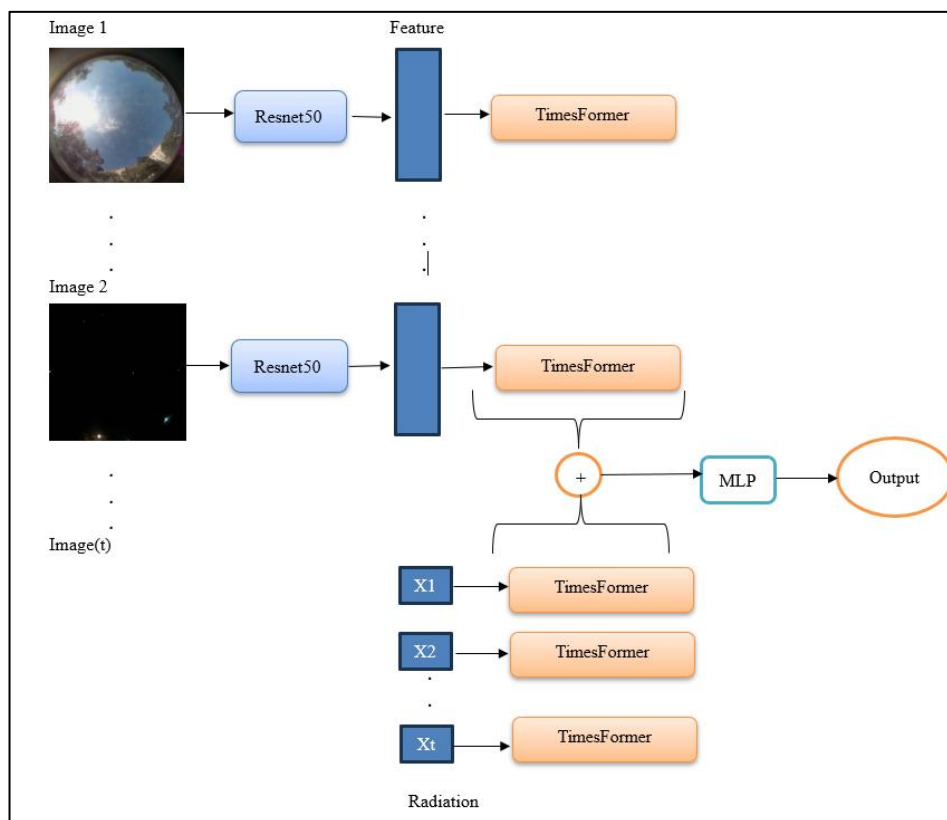
Step 2: Các đặc trưng hình ảnh sau đó được đưa vào Transformer để xử lý dữ liệu chuỗi các đặc trưng ảnh theo thời gian

Step 3: Dữ liệu bức xạ đầu vào cũng được đưa vào Transformer riêng biệt để mô hình học và nắm bắt các mẫu hoặc xu hướng trong dữ liệu bức xạ theo thời gian.

Step 4: Đầu ra từ cả hai nhánh Transformer (từ ảnh và từ bức xạ) được Concat lại để kết hợp thông tin từ cả hai nguồn dữ liệu.

Step 5: Kết quả tổng hợp này sau đó được đưa vào một mạng Multi-layer Perceptron để xử lý các đặc trưng kết hợp này sau đó đưa ra đầu ra cuối cùng của mô hình.

4.3 Mô hình Resnet kết hợp với Timesformer



Hình 4.1: Mô hình Resnet+TimesFormer

- Cách hoạt động của mô hình:

Step 1: Mỗi ảnh đầu vào trong chuỗi thời gian được đưa qua một mạng ResNet50 đã pretrained (bỏ lớp fully-connected) để trích xuất đặc trưng hình ảnh. Kết quả là một chuỗi vector đặc trưng.

Step 2: Chuỗi đặc trưng ảnh này sau đó được đưa vào khối mã hóa TimeSformer, một dạng Transformer cho dữ liệu chuỗi thời gian, để học mối quan hệ không gian - thời gian giữa các khung hình. TimeSformer học attention qua các vị trí không gian (ảnh) và các thời điểm khác nhau (chuỗi ảnh).

Step 3: Đồng thời, chuỗi dữ liệu bức xạ đầu vào theo thời gian cũng được đưa vào một nhánh mã hóa riêng để trích xuất mẫu bức xạ theo thời gian.

Step 4: Đầu ra từ TimeSformer (đặc trưng từ ảnh) và đầu ra từ nhánh xử lý dữ liệu bức xạ sẽ được gộp lại (concatenate), tạo thành một vector tổng hợp bao gồm cả thông tin từ ảnh và dữ liệu vật lý.

Step 5: Kết quả sau đó được đưa qua một mạng MLP (Multi-Layer Perceptron), bao gồm một hoặc nhiều tầng fully-connected, để xử lý và sinh ra dự đoán cuối cùng về giá trị bức xạ tại thời điểm tương lai.

Chương V. Kết quả và đánh giá

5.1 Cấu hình huấn luyện và các độ đo đánh giá

5.1.1 Cấu hình huấn luyện

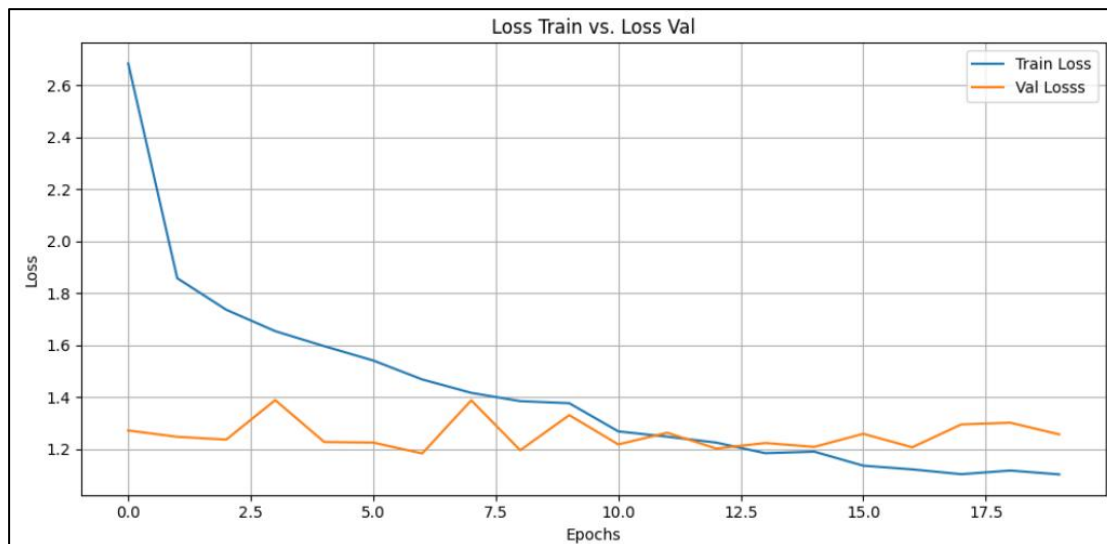
- Các mô hình đề xuất sử dụng kiến trúc mạng backbone sử dụng là ResNet50. Ta thực hiện huấn luyện mô hình bằng bộ tối ưu hóa Adam với tốc độ học là $1e-4$, sử dụng hàm mất mát MSE với tổng số epoch huấn luyện là 20 epochs. Mô hình sẽ thực hiện dự đoán sau thời gian là 5 phút Bên cạnh đó, batch_size train là 8, mô hình được huấn luyện trên GPU P100 (kaggle)

5.1.2 Các độ đo đánh giá

Ta sử dụng các chỉ số đánh giá như: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) và R-squared (R^2)

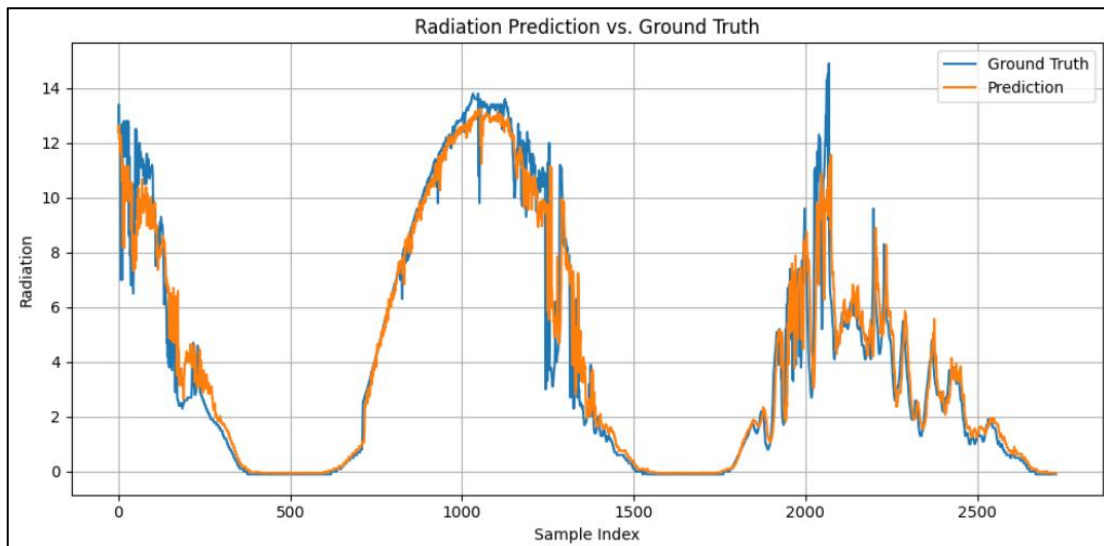
5.2 Kết quả thực nghiệm

5.2.1 Với mô hình Resnet+LSTM



Hình 5.2.1.1: Loss của mô hình trên 2 tập Train và Val

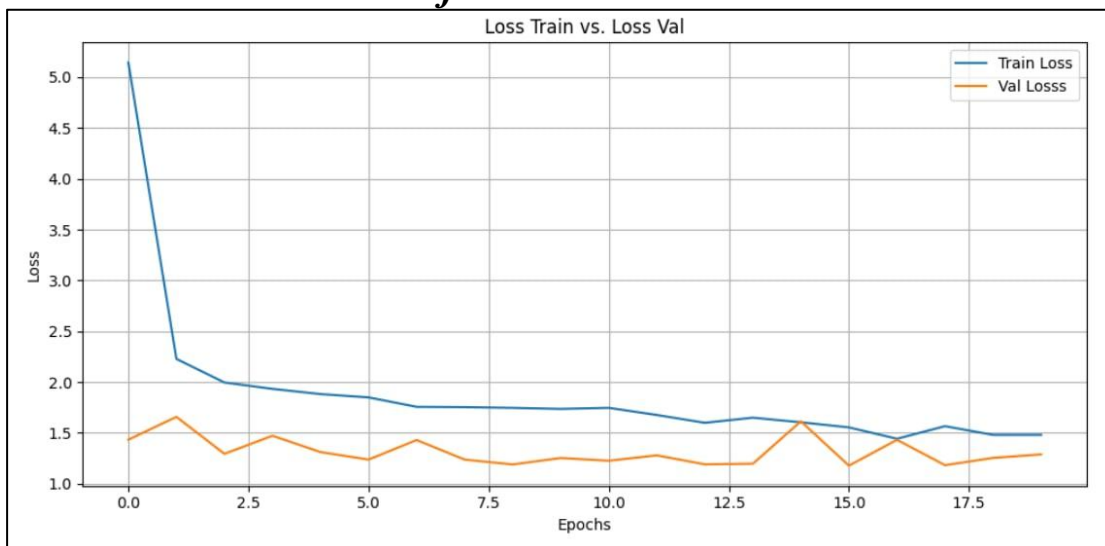
- Nhận xét: Cả Train Loss và Val Loss đều có xu hướng giảm dần theo số lượng epochs, điều này cho thấy mô hình đang học và cải thiện hiệu suất. Tuy nhiên từ hình trên nên cần cải thiện thêm về mô hình để Loss của mô hình có thể giảm thêm nữa và nên áp dụng kỹ thuật dừng khi mô hình đã hội tụ ở epoch thứ 11



Hình 5.2.1.2: So sánh giữa giá trị thực tế và giá trị dự đoán của mô hình

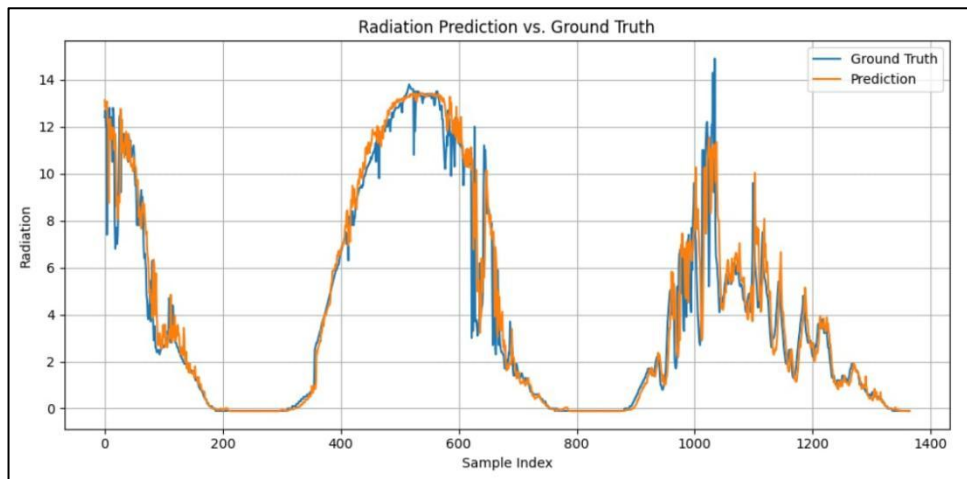
- Nhận xét: Biểu đồ trên cho thấy giá trị dự đoán có xu hướng bám khá sát đường thực tế. Tuy nhiên, vẫn còn một số sai lệch nhất định, đặc biệt là ở các thời điểm có sự thay đổi nhanh hoặc giá trị bức xạ cực đại.

5.2.2 Với mô hình Resnet+Transformer



Hình 5.2.2.1: Loss của mô hình trên 2 tập Train và Val

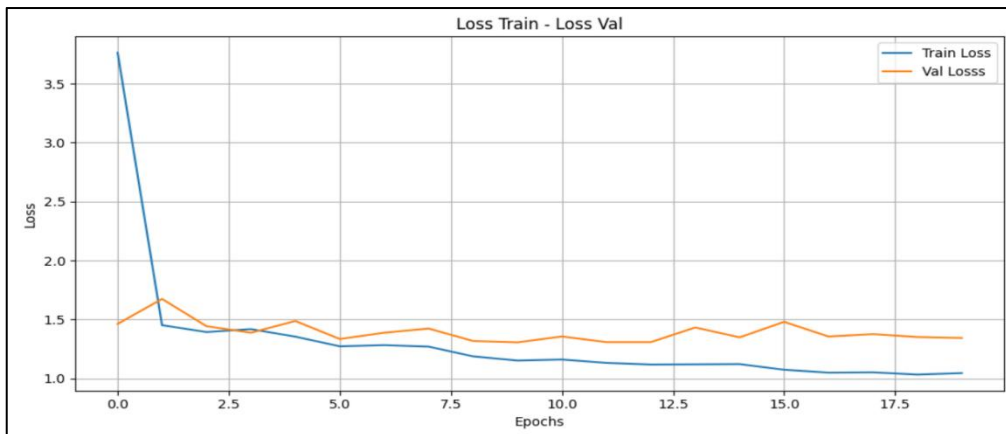
- Nhận xét: Cả Train Loss và Val Loss có xu hướng ổn định, nhưng chưa giảm nhiều ở cuối → Có thể cần thêm epoch để hội tụ tốt hơn, điều này cho thấy mô hình đang học và cải thiện hiệu suất. Tuy nhiên từ hình trên nên cần cải thiện thêm về mô hình để Loss của mô hình có thể giảm thêm nữa



Hình 5.2.2.2: So sánh giữa giá trị thực tế và giá trị dự đoán của mô hình

- Nhận xét: Biểu đồ trên cho thấy giá trị dự đoán có xu hướng bám khá sát đường thực tế và mô hình nắm rõ biến động theo thời gian, đặc biệt các pha tăng – giảm mạnh. Tuy nhiên, vẫn còn một số sai lệch nhất định, đặc biệt là ở các thời điểm có sự thay đổi nhanh hoặc giá trị bức xạ cực đại

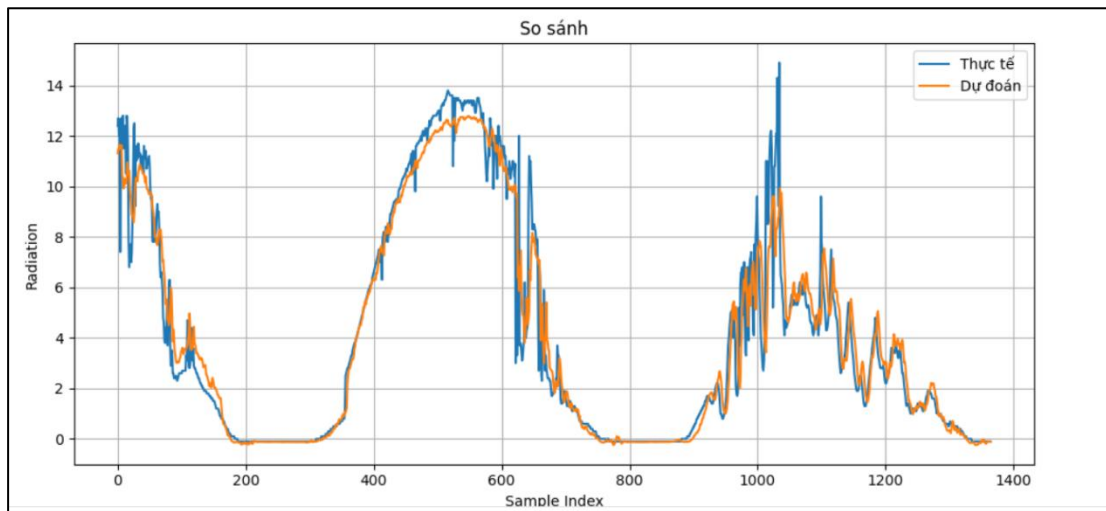
5.2.3 Với mô hình Resnet+TimesFormer



Hình 5.2.3.1: Loss của mô hình trên 2 tập Train và Val

*Nhận xét:

- Loss huấn luyện (Train Loss) giảm rất nhanh trong vài epoch đầu, từ ~4 xuống dưới 1.5, sau đó tiếp tục giảm chậm dần.
 - Loss kiểm tra (Validation Loss) dao động nhẹ quanh giá trị 1.3–1.5 từ epoch 3 trở đi, không giảm đều như train loss.
 - Hiện tượng overfitting nhẹ có thể bắt đầu sau khoảng epoch 5: train loss tiếp tục giảm, trong khi val loss không cải thiện rõ rệt, thậm chí dao động tăng giảm.
- Mô hình học nhanh và ổn định. Tuy nhiên, có dấu hiệu overfitting nhẹ do khoảng cách giữa train và val loss bắt đầu giãn ra sau vài epoch



Hình 5.2.3.2: So sánh giữa giá trị thực tế và giá trị dự đoán của mô hình

***Nhận xét:**

- Đường dự đoán (cam) bám sát khá tốt đường thực tế (xanh), đặc biệt là những phần mượt mà theo chu kỳ ngày - đêm.
 - Ở các vùng có biến động nhanh hoặc đỉnh nhọn mô hình chưa bắt kịp được biên độ biến động.
- Mô hình thể hiện hiệu quả dự đoán tốt trên phần lớn chuỗi thời gian, đặc biệt ở vùng ổn định. Tuy nhiên, vẫn còn sai lệch ở các đoạn có biến động mạnh.

5.3 So sánh giữa các mô hình

Mô hình	MAE	RMSE	R^2
Resnet+LSTM	0.6007	1.1095	93.34%
Resnet+Transformer	0.5922	1.1352	93.74%
Resnet+TimesFormer	0.6784	1.1584	93.11%

Hình 5.3: Bảng so sánh giữa các mô hình

Nhận xét:

- ResNet + Transformer đạt R^2 cao nhất (93.74%), cho thấy mô hình học tốt mối quan hệ giữa ảnh và chuỗi thời gian, dù RMSE hơi cao hơn so với ResNet+LSTM.
- ResNet + LSTM có RMSE thấp nhất (1.1095) và MAE khá thấp, cho thấy dự đoán ổn định và ít sai lệch tuyệt đối hơn, mặc dù R^2 chỉ xếp thứ hai.
- ResNet + TimeSformer có độ chính xác tổng thể thấp hơn hai mô hình còn lại ($R^2 = 93.11\%$), MAE và RMSE cao nhất, cho thấy mô hình chưa khai thác tốt được đặc trưng thời gian từ ảnh chuỗi trong bài toán.

Chương VI. Kết luận và hướng phát triển

6.1 Kết luận

- Điểm đạt được:

- + Xây dựng được Pipeline mô hình dự báo bức xạ mặt trời kết hợp ảnh Sky Cam và dữ liệu bức xạ.
- + Mô hình kết hợp (Resnet+TimesFormer) cho độ chính xác cao hơn
- + Hiệu quả mô hình được đánh giá qua các chỉ số MAE, RMSE, R^2 với kết quả khả quan.

- Điểm chưa đạt được:

- + Chưa tối ưu được tốt sai số của mô hình. Kết quả mới ở mức chấp nhận được
- + Bộ dữ liệu còn giới hạn về quy mô
- + Thiếu đánh giá mô hình theo các mùa hoặc điều kiện thời tiết khác nhau.

6.2 Hướng phát triển

- Mở rộng mô hình để dự báo thời gian thực (real-time).
- Tăng độ chính xác bằng cách sử dụng mô hình học sâu nâng cao như Vision Transformer (ViT), ConvLSTM.
- Thu thập thêm dữ liệu dài hạn, đa dạng điều kiện thời tiết và vị trí địa lý.
- Phát triển giao diện hiển thị kết quả dự báo trực quan (dashboard web hoặc mobile).

TÀI LIỆU THAM KHẢO

- [1] <https://lctech.vn/nang-luong-tai-ao-chia-khoa-phat-trien-ben-vung-trong-the-ky-21/>
- [2] <https://www.kaggle.com/datasets/alenjako/li/sky-images-and-solar-radiation-measurement-dataset>
- [3] <https://aicandy.vn/mo-hinh-resnet-dot-pha-trong-nhan-dien-hinh-anh/>
- [4] Duffie, J.A., & Beckman, W.A. (2013). Solar Engineering of Thermal Processes (4th ed.). Wiley.
- [5] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
Brownlee, J. (2017). Deep Learning for Time Series Forecasting. Machine Learning Mastery.
- [6] Yang, D., & Bright, J. M. (2020). "Very short-term irradiance forecasting using sky images and deep learning." Renewable Energy, 145, 874–884.
- [7] Kim, D., & Han, G. (2019). "Short-term solar irradiance prediction based on sky images using deep learning." Energies, 12(7), 1302.