

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA ĐIỆN

ĐỒ ÁN TỐT NGHIỆP
CAPSTONE PROJECT

NGÀNH: KỸ THUẬT ĐIỀU KHIỂN VÀ TỰ ĐỘNG HÓA

ĐỀ TÀI:

**ỨNG DỤNG EDGE AI ĐỂ DỰ BÁO SỚM LŨ LỤT
ĐÔ THỊ**

Người hướng dẫn: **TS. NGÔ ĐÌNH THANH**

Cán bộ hướng dẫn: **KS. TRỊNH THẾ TRƯỜNG**

Sinh viên thực hiện:

1. NGUYỄN THÀNH ĐẠT – 105200398 – 20TDHCLC4

2. NGUYỄN TIẾN AN – 105200398 – 20TDHCLC2

3. NGUYỄN TRỌNG THANH – 105200429 – 20TDHCLC2

Đà Nẵng, 6/2025

TÓM TẮT

Tên đề tài: **Ứng dụng Edge AI để dự báo sớm lũ lụt đô thị**

Sinh viên thực hiện 1: Nguyễn Tiên An

Số thẻ sinh viên 1: 105200398

Lớp: 20TDHCLC2

Sinh viên thực hiện 2: Nguyễn Thành Đạt

Số thẻ sinh viên 2: 105200487

Lớp: 20TDHCLC4

Sinh viên thực hiện 3: Nguyễn Trọng Thanh

Số thẻ sinh viên 3: 105200429

Lớp: 20TDHCLC2

Đề tài này tập trung nghiên cứu và triển khai một hệ thống dự báo lũ lụt đô thị ứng dụng trí tuệ nhân tạo (AI) kết hợp với kiến trúc Edge-Cloud. Mục tiêu chính của đề tài là xây dựng một giải pháp dự báo mực nước theo thời gian thực nhằm hỗ trợ công tác cảnh báo sớm tại các khu vực thường xuyên xảy ra ngập úng ở thành phố Đà Nẵng.

Nhóm tác giả sử dụng mô hình học sâu MTS-LSTM để xử lý chuỗi thời gian nhiều biến đầu vào (lượng mưa, mực nước, các yếu tố khí tượng thủy văn), được huấn luyện trên tập dữ liệu thực tế từ năm 2023–2024 tại Đà Nẵng. Quá trình huấn luyện và thử nghiệm được thực hiện trên Google Colab với các kỹ thuật tiền xử lý, tối ưu mô hình và đánh giá bằng các chỉ số như $NSE = 0,73$ và hệ số tương quan $r = 0,86$. Mô hình cho thấy khả năng dự báo hiệu quả các đỉnh lũ và độ trễ ngắn, phù hợp với yêu cầu cảnh báo sớm.

Hệ thống được thiết kế với kiến trúc phân tán: dữ liệu được thu thập từ cảm biến tại các trạm đo (giả lập bằng file .csv), sau đó được xử lý sơ bộ tại thiết bị Edge (mô phỏng vi điều khiển EPS32). Dữ liệu sau xử lý được truyền lên Cloud để đưa vào mô hình AI nhằm đưa ra dự báo, sau đó hiển thị kết quả mực nước dự báo. Việc mô phỏng toàn bộ hệ thống thay vì triển khai phần cứng giúp tập trung vào phát triển thuật toán lõi và đảm bảo tính khả thi trong môi trường giới hạn về thời gian và tài nguyên.

Hệ thống Edge AI này giúp giảm độ trễ xử lý từ hàng giây xuống còn khoảng 100 mili-giây, tiết kiệm băng thông truyền tải dữ liệu đến 97%, đồng thời tăng cường tính bảo mật nhờ xử lý cục bộ tại thiết bị biên. Kết quả mô phỏng cho thấy kiến trúc có thể mở rộng trong tương lai, dễ tích hợp vào các nền tảng đô thị thông minh như Danang Smart City.

Tất cả các phân tử tiền xử lý dữ liệu, huấn luyện mô hình, mô phỏng truyền dữ liệu đến hiển thị kết quả và cảnh báo đều được tích hợp liền mạch trong một pipeline chạy trên môi trường mô phỏng, nhưng được thiết kế linh hoạt để dễ dàng gắn kết với phần cứng thực tế trong các bước tiếp theo.

KHOA ĐIỆN

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

TT	Họ tên sinh viên	Số thẻ SV	Lớp	Ngành
1	Nguyễn Thành Đạt	105200487	20TDHCLC4	Kỹ thuật điều khiển tự động hóa
2	Nguyễn Tiến An	105200398	20TDHCLC2	Kỹ thuật điều khiển tự động hóa
3	Nguyễn Trọng Thanh	105200429	20TDHCLC2	Kỹ thuật điều khiển tự động hóa

1. Tên đề tài đồ án:

Ứng dụng Edge AI để dự báo sớm lũ lụt đô thị

2. Đề tài thuộc diện: Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện

3. Các số liệu và dữ liệu ban đầu:

- Dữ liệu khí tượng – thủy văn theo ngày tại thành phố Đà Nẵng
- Dữ liệu khí tượng – thủy văn theo giờ tại thành phố Đà Nẵng
- Dữ liệu mực nước – lượng mưa tại trạm đo khí tượng tại hồ Thạc Gián, thành phố Đà Nẵng
- Bản đồ địa hình số (DEM) khu vực sông Cẩm Lệ, thành phố Đà Nẵng

4. Nội dung các phần thuyết minh và tính toán:

a. Phần chung:

TT	Họ tên sinh viên	Nội dung
1	Nguyễn Tiến An	Nghiên cứu tổng quan tình trạng ngập lụt tại Đà Nẵng Nghiên cứu tổng quan về các thuật toán dự đoán lũ lụt Thực nghiệm hệ thống mô phỏng, đánh giá kết quả, phương hướng phát triển và kết luận Viết báo cáo thuyết minh
2	Nguyễn Thành Đạt	
3	Nguyễn Trọng Thanh	

b. Phần riêng:

TT	Họ tên sinh viên	Nội dung
1	Nguyễn Thành Đạt	Nghiên cứu và triển khai mô hình học máy MTS-LSTM để dự đoán mực nước

		Tiền xử lý dữ liệu đầu vào cho mô hình (chuẩn hóa, tạo sliding window, chia train/test) Huấn luyện, đánh giá và tinh chỉnh mô hình Viết báo cáo chương 2, 4
2	Nguyễn Tiến An	Nghiên cứu kiến trúc hệ thống Edge AI và Cloud AI trong bối cảnh dự báo lũ đô thị Phân tích và lựa chọn phần cứng phù hợp cho thiết bị biên Xây dựng mô hình mô phỏng hệ thống và ước tính hiệu năng Viết báo cáo chương 1, 2, 3
3	Nguyễn Trọng Thanh	Thu thập và xử lý dữ liệu cảm biến mực nước và mưa (định dạng CSV, làm sạch dữ liệu) Kết nối dữ liệu từ thiết bị biên đến cloud và hệ thống mô hình AI Viết báo cáo chương 1

5. Các bản vẽ, đồ thị::

.....

.....

6. <i>Họ tên người hướng dẫn:</i>	<i>Phân/ Nội dung:</i>
TS. Ngô Đình Thanh	Theo dõi, hướng dẫn và đóng góp ý kiến trong quá trình thực hiện đề tài
KS. Trịnh Thế Trường	Cung cấp thông tin, tài liệu, số liệu liên quan đến đề tài

7. Ngày giao nhiệm vụ đồ án: 24/02/2025

8. Ngày hoàn thành đồ án: 17/06/2025

Đà Nẵng, ngày tháng 6 năm 2025

Trưởng Bộ môn Tự động hóa

Người hướng dẫn

TS. Giáp Quang Huy

TS. Ngô Đình Thanh

PHIẾU KIỂM SOÁT TIẾN ĐỘ LÀM ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên:

Nguyễn Tiến An

Số thẻ SV: 105200398

Nguyễn Thành Đạt

Số thẻ SV: 105200487

Nguyễn Trọng Thanh

Số thẻ SV: 105200429

Tên đề tài ĐATN: **Ứng dụng Edge AI để dự báo sớm lũ lụt đô thị**

Họ tên người HD: TS. Ngô Đình Thanh, Khoa Điện, Trường Đại học Bách Khoa – Đại học Đà Nẵng

Tuần	Ngày	Khối lượng		GVHD ký tên
		đã thực hiện (%)	tiếp tục thực hiện (%)	
1	03/03/2025	Nhận đề tài	Gặp giảng viên hướng dẫn và trao đổi với đại diện doanh nghiệp	
2	10/03/2025	Tìm hiểu các kiến thức liên quan, đọc các nghiên cứu trên các bài báo khoa học	Báo cáo hoạt động với đại diện doanh nghiệp	
3	17/03/2025	Đọc các nghiên cứu trên các bài báo khoa học	Viết báo cáo chương 1	
4	24/03/2025	Duyệt lần 1: Đánh giá khối lượng hoàn thành _____ % : Được tiếp tục làm ĐATN <input type="checkbox"/> Không tiếp tục thực hiện ĐATN <input type="checkbox"/>		
5	31/03/2025	Tìm hiểu về ngôn ngữ lập trình Python	Tìm hiểu ngôn ngữ lập trình Python	
6	07/04/2025	Nắm được cơ bản ngôn ngữ lập trình Python	Tìm hiểu ngôn ngữ lập trình Python	
7	14/04/2025	Tổng hợp các trường dữ liệu liên quan đến đề tài và xin dữ liệu từ doanh nghiệp	Xin dữ liệu từ doanh nghiệp và tìm hiểu các mô hình thuật toán AI dự đoán	

8	21/04/2025	Duyệt lần 2: Đánh giá khối lượng hoàn thành _____ % : Được tiếp tục làm ĐATN <input type="checkbox"/> Không tiếp tục thực hiện ĐATN <input type="checkbox"/>	
9	28/04/2025	Xin dữ liệu từ doanh nghiệp và tìm hiểu các mô hình thuật toán dự đoán và lựa chọn mô hình thuật toán dự đoán	Nghiên cứu về mô hình thuật toán MTS - LSTM
10	05/05/2025	Nhận được dữ liệu và tiến hành tiền xử lý dữ liệu và training dữ liệu trên Colab	Lựa chọn phương án thiết kế hệ thống mô phỏng thiết bị biên, chứng minh các sự cần thiết sử dụng thiết bị biên
11	12/05/2025	Nghiên cứu cách tiền xử lý dữ liệu trên thiết bị biên	Viết báo cáo chương 3
12	19/05/2025	Duyệt lần 3: Đánh giá khối lượng hoàn thành _____ % : Được tiếp tục làm ĐATN <input type="checkbox"/> Không tiếp tục thực hiện ĐATN <input type="checkbox"/>	
13	26/05/2025	Tiến hành thực nghiệm hệ thống mô phỏng	Hiệu chỉnh đánh giá kết quả Viết báo cáo chương 4
14	02/06/2025	Tiến hành kiểm nghiệm và hiệu chỉnh, đánh giá kết quả	Tổng hợp tài liệu, hoàn thiện báo cáo theo mẫu
15	09/06/2025	Viết thuyết minh, hoàn thiện đề tài	In thuyết minh

LỜI NÓI ĐẦU VÀ CẢM ƠN

Hiện nay, khoa học và công nghệ đang phát triển mạnh mẽ, mang lại nhiều lợi ích thiết thực cho đời sống con người cả về vật chất lẫn tinh thần. Trong bối cảnh biến đổi khí hậu ngày càng phức tạp, Nhà nước đang đẩy mạnh ứng dụng công nghệ cao vào quản lý và phòng chống thiên tai, đặc biệt là lũ lụt – một trong những hiểm họa ngày càng gia tăng ở các đô thị ven biển như Đà Nẵng. Cùng với sự phát triển của trí tuệ nhân tạo (AI), các mô hình học máy đang được ứng dụng rộng rãi trong dự báo thủy văn. Tuy nhiên, việc triển khai AI trên nền tảng điện toán đám mây truyền thống vẫn còn nhiều hạn chế, đặc biệt là độ trễ, chi phí truyền tải dữ liệu và vấn đề riêng tư. Trước thực tế đó, **Edge AI – trí tuệ nhân tạo tại thiết bị biên** đang nổi lên như một giải pháp tiềm năng, cho phép xử lý dữ liệu tại chỗ, nhanh chóng và hiệu quả hơn trong các hệ thống cảnh báo sớm. Nhận thấy được tính cấp thiết và tiềm năng ứng dụng của công nghệ mới này, nhóm tác giả đã lựa chọn đề tài: “**Ứng dụng Edge AI để dự báo sớm lũ lụt đô thị**” làm hướng nghiên cứu cho đề án tốt nghiệp.

Trong suốt quá trình học tập và thực hiện đề án tốt nghiệp em luôn được sự quan tâm, hướng dẫn và giúp đỡ tận tình của các thầy, cô giáo trong Khoa Điện cùng với sự động viên giúp đỡ của bạn bè đồng nghiệp.

Lời đầu tiên em xin được bày tỏ lòng biết ơn sâu sắc đến Ban giám hiệu Trường Đại học Bách Khoa - Đại học Đà Nẵng và Ban chủ nhiệm Khoa Điện đã tận tình giúp đỡ cho em suốt thời gian học tại trường.

Em xin cảm ơn KS. Trịnh Thế Trường tại công ty Cổ phần Tư vấn và Phát triển Kỹ thuật Tài nguyên nước đã truyền đạt những kiến thức hữu ích để em có thể hiểu rõ hơn về lĩnh vực đang nghiên cứu.

Đặc biệt em xin bày tỏ lòng biết ơn chân thành sâu sắc tới thầy giáo TS. Ngô Đình Thanh đã trực tiếp giúp đỡ, hướng dẫn em hoàn thành báo cáo này.

Tuy nhiên do kiến thức còn hạn hẹp, chưa tiếp xúc nhiều với thực tiễn cũng như các tài liệu tham khảo còn quá ít, trong khi đó thời gian thực hiện cũng có hạn nên trong đồ án không tránh khỏi những thiếu sót, rất mong nhận được những lời chỉ dẫn thêm từ các thầy, cô.

Cuối cùng em kính chúc quý thầy, cô dồi dào sức khỏe và thành công trong sự nghiệp cao quý. Đồng kính chúc các cô, chú, anh, chị trong công ty TNHH THTECH nơi em thực hiện đề án tốt nghiệp luôn dồi dào sức khỏe, đạt được nhiều thành công trong công việc.

LỜI CAM ĐOAN LIÊM CHÍNH HỌC THUẬT

Tôi xin cam đoan đây là công trình nghiên cứu của tôi và được sự hướng dẫn khoa học của TS. Ngô Đình Thanh, KS. Trịnh Thế Trường. Các nội dung nghiên cứu trong đề tài “Ứng dụng Edge AI để dự báo sớm lũ lụt đô thị” của tôi là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng thống kê phục vụ cho việc phân tích, nhận xét, đánh giá được cá nhân thu thập từ các nguồn khác nhau có ghi rõ nguồn gốc. Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung bài báo cáo của mình. Sinh viên thực hiện

Sinh viên thực hiện

MỤC LỤC

Tóm tắt	i
Nhiệm vụ đề án	ii
Kiểm soát tiến độ làm đề án tốt nghiệp	iv
Lời nói đầu và cảm ơn	vi
Lời cam đoan liêm chính học thuật	vii
Mục lục	viii
Danh sách các bảng biểu, hình vẽ và sơ đồ	x
Danh sách các cụm từ viết tắt	xi
MỤC LỤC	viii
DANH SÁCH CÁC BẢNG, HÌNH VẼ.....	xi
DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT.....	xii
MỞ ĐẦU	1
1. Đặt vấn đề.....	1
2. Mục tiêu đề tài.....	2
3. Phạm vi và đối tượng nghiên cứu	2
4. Phương pháp nghiên cứu.....	2
5. Cấu trúc đề án tốt nghiệp	3
Chương 1: TỔNG QUAN VỀ ĐỀ TÀI.....	4
1.1. Lý do chọn đề tài	4
1.2. Hệ thống quan trắc ngập lụt tại Đà Nẵng	6
1.2.1. Hiện trạng và giải pháp lũ lụt hiện nay tại thành phố Đà Nẵng.....	6
1.2.2. Phân cứng của các thiết bị điều khiển chính giám sát thu thập dữ liệu	7
1.3. Các nghiên cứu trước đó.....	9
1.4. Kết luận chương	10
Chương 2: CƠ SỞ KHOA HỌC	12

2.1.	Dữ liệu Khí tượng – Thủy văn tại Đà Nẵng	12
2.1.1.	Dữ liệu theo ngày (Daily):.....	14
2.1.2.	Dữ liệu theo giờ (Hourly):.....	15
2.2.	Tiền xử lý dữ liệu.....	16
2.3.	Chuẩn hóa dữ liệu (Normalization)	18
2.4.	Lựa chọn mô hình AI.....	19
2.4.1.	Mạng Nơ-ron hồi tiếp LSTM (Long Short Term Memory)	19
2.4.2.	Ưu điểm của LSTM so với RNN truyền thống.....	22
2.5.	Mô hình LSTM đa thang thời gian (MTS - LSTM)	24
2.5.1.	Kiến trúc tổng quan của mô hình MTS – LSTM.....	24
2.5.2.	Cơ chế truyền trạng thái giữa các nhánh LSTM	25
2.5.3.	Đầu vào và đầu ra của từng nhánh LSTM	25
2.5.4.	Ưu điểm của mô hình MTS – LSTM so với LSTM thông thường	27
2.6.	Mô hình truyền thống Cloud AI.....	28
Chương 3: EDGE AI – HƯỚNG TIẾP CẬN MỚI TRONG DỰ BÁO VÀ CẢNH BÁO LŨ LỤT		29
3.1.	Lí do ứng dụng Edge AI trong bối cảnh lũ lụt đô thị.....	29
3.2.	So sánh Edge AI với mô hình tập trung truyền thống	29
3.2.1.	Ưu điểm của Edge AI	29
3.2.2.	So sánh định lượng qua các chỉ số	31
3.2.3.	Kết luận	34
3.3.	Lựa chọn thiết bị Edge AI phù hợp.....	34
3.4.	Thiết kế tổng quan hệ thống Edge AI trong dự báo lũ lụt	35
3.4.1.	Kiến trúc tổng thể hệ thống.....	35
3.4.2.	Sơ đồ mô hình kiến trúc phân tán Edge – Cloud	36
3.4.3.	Sơ đồ luồng dữ liệu trong mô hình mô phỏng	36
3.5.	Triển khai hệ thống Edge AI.....	37
3.5.1.	Lý do không triển khai phần cứng	37
3.5.2.	Kiến trúc mô phỏng hệ thống Edge AI.....	38

3.5.3.	So sánh giữa mô phỏng và phân cứng	39
3.5.4.	Tính mở rộng về sau:	40
3.5.5.	Kết luận	40
Chương 4: KẾT QUẢ HUẤN LUYỆN VÀ KIỂM TRA MÔ HÌNH MTS - LSTM....		41
4.1.	Bộ chỉ số đánh giá hiệu năng mô hình	41
4.2.	Phân tích kết quả dự báo và hiệu suất mô hình	48
4.3.	Ưu điểm và hạn chế của mô hình MTS - LSTM.....	52
4.3.1.	Ưu điểm.....	52
4.3.2.	Hạn chế.....	52
4.4.	Mô phỏng ngập lụt khu vực nghiên cứu.....	53
4.5.	Tổng kết.....	55
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....		57
TÀI LIỆU THAM KHẢO		59
PHỤ LỤC 1		Error! Bookmark not defined.
PHỤ LỤC 2		Error! Bookmark not defined.

DANH SÁCH CÁC BẢNG, HÌNH VẼ

Danh sách các bảng

Bảng 3.1. So sánh kiểm nghiệm trên mô phỏng và phần cứng.....	39
Bảng 4.1. Hiệu năng mô hình MTS-LSTM trên tập huấn luyện, xác thực và kiểm thử (các chỉ số chính).....	45

Danh sách Hình vẽ

Hình 1.1. Bản đồ mức mưa, ngập nước tại thành phố Đà Nẵng (Nguồn: app Smart City)	7
Hình 1.2. Trạm đo mưa tự động Watec	8
Hình 1.3. Trạm đo mực nước tự động Watec	9
Hình 2.1. Dữ liệu khí tượng – Thủy văn theo ngày tại thành phố Đà Nẵng	14
Hình 2.2. Dữ liệu khí tượng thủy văn theo giờ tại thành phố Đà Nẵng	15
Hình 2.3. Biểu đồ thống kê Lượng mưa và Mực nước trung bình tại Cẩm Lệ - Đà Nẵng (năm 2023).....	16
Hình 2.4. Kiến trúc mô hình LSTM đa thang thời gian (MTS - LTSM).....	24
Hình 2.5. Sơ đồ mô hình kiến trúc truyền thống Cloud AI	28
Hình 3.1. Lựa chọn thiết bị Edge.....	35
Hình 3.2. Sơ đồ mô hình kiến trúc phân tán Edge – Cloud.....	36
Hình 3.3. Sơ đồ luồng dữ liệu của hệ thống Edge AI trong mô hình mô phỏng.....	37
Hình 3.4. Lưu đồ thuật toán tiền xử lý dữ liệu tại thiết bị biên.....	39
Hình 4.1. So sánh mực nước trung bình ngày dự báo bởi MTS – LSTM và mực nước thực tế tại trạm thủy văn Cẩm Lệ, Đà Nẵng (07 – 12/2023).....	48
Hình 4.2. Diễn biến sai số dự báo mực nước theo thời gian trên tập kiểm thử.....	50
Hình 4.3. Biểu đồ phân tán giữa mực nước quan trắc và mực nước dự báo (bình quân ngày) trên tập kiểm thử.....	51
Hình 4.4. Bản đồ địa hình số (DEM) khu vực sông Cẩm Lệ, Đà Nẵng.....	54
Hình 4.5. Bản đồ ngập lụt dựa trên mực nước 1.5m.	55

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

KÝ HIỆU:

X'	Chuẩn hóa Min - Max
x''	Chuẩn hóa Z – score
μ	Giá trị trung bình của biến
σ	Độ lệch chuẩn của biến
FC_c	Tầng fully connected xử lý vector context đầu vào
FC_h	Tầng fully connected xử lý vector hidden state đầu ra của LSTM.
FC^D	Tầng fully connected dùng cho nhánh dữ liệu theo ngày.
FC^H	Tầng fully connected dùng cho nhánh dữ liệu theo giờ.
T^D	Số bước thời gian trong chuỗi dữ liệu theo ngày
h^D	Trạng thái ẩn cuối cùng từ nhánh LSTM ngày
c^D	Trạng thái bộ nhớ cuối cùng từ nhánh LSTM ngày
T^H	Số bước thời gian trong chuỗi dữ liệu theo giờ
x_d^D	vector đầu vào ngày d
h_d^D	Trạng thái ẩn cho nhánh LSTM giờ (sau khi ánh xạ từ h^D)
c_d^D	Trạng thái bộ nhớ cho nhánh LSTM giờ (sau khi ánh xạ từ c^D)
$H_{T^D}^D$	Trạng thái cuối chưa thông tin tổng hợp
y^D, y^H	Giá trị mực nước dự báo ở thang thời gian ngày và giờ
W_D, W_H	Ma trận trọng số cho thời gian ngày và giờ
b^D	Hệ số điều chỉnh (bias) của lớp đầu ra cho nhánh ngày
X_t^H	Vecto đặc trưng tại giờ t
h_0^D	Trạng thái ẩn khởi tạo cho nhánh giờ
c_0^D	Trạng thái bộ nhớ khởi tạo cho nhánh giờ
T_{upload}	Thời gian truyền dữ liệu từ trạm đến server
$T_{inference}$	Thời gian server xử lý mô hình AI
$T_{download}$	Thời gian phản hồi kết quả về trạm
n	Số mẫu dữ liệu gửi mỗi ngày
s	Kích thước mỗi mẫu
B_{cloud}	Lượng băng thông cần thiết để truyền toàn bộ dữ liệu

E	Năng lượng tiêu thụ khi truyền dữ liệu
V	Điện áp (Voltage)
R	Điện trở (Resistance)
t	Thời gian truyền nhận dữ liệu
C_{cloud}	Tổng chi phí vận hành của Cloud AI
C_{server}	Chi phí phần cứng/máy chủ của Cloud
$C_{compute}$	Chi phí tính toán (computational cost) của Cloud
$C_{storage}$	Chi phí lưu trữ dữ liệu trong Cloud
$C_{network}$	Chi phí truyền tải dữ liệu mạng trong Cloud
C_{edge}	Tổng chi phí vận hành của Edge AI
$C_{element}$	Chi phí phần tử triển khai (thiết bị tại điểm nút – ví dụ cảm biến thông minh, bo mạch...)
$C_{maintain}$	Chi phí bảo trì sửa chữa thiết bị Edge sau khi triển khai
H_{obs}	Giá trị quan trắc thực tế
H_{sim}	Giá trị mô phỏng bởi mô hình tại các thời điểm $t = 1, 2, \dots, N$
\bar{H}_{obs}	Giá trị trung bình của H_{obs} trên tập dữ liệu đang xét
r	Hệ số tương quan Pearson giữa H_{sim} và H_{obs} ;
$\alpha = \sigma_{sim} / \sigma_{obs}$	Tỷ số độ lệch chuẩn (đại diện cho sai khác về độ biến động – còn gọi là Alpha-NSE);
$\beta_{KGE} = \mu_{sim} / \mu_{obs}$	Tỷ số trung bình mô phỏng so với trung bình quan trắc (đại diện cho sai khác về độ lệch hệ thống – còn gọi là BetaKGE).
$H_{x,h}$	Giá trị dòng chảy (hoặc mực nước) tương ứng vị trí thứ h
$H_{x,p}$	Lưu lượng (hoặc mực nước) tại phân vị p
$H_{x,l}$	Giá trị dòng chảy ở vị trí thứ l
$H^{(peak_i)}$	Giá trị mực nước đỉnh lũ thứ i (mô phỏng hoặc quan trắc)

CHỮ VIẾT TẮT:

Mô hình và công nghệ

LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
XgBoost	Extreme Gradient Boosting
MTS – LSTM	Multivariate Time Series Long Short-Term Memory (LSTM cho chuỗi thời gian đa biến)
ANN	Artificial Neural Network
CNN	Convolutional Neural Network

Edge AI Artificial Intelligence at the Edge (Trí tuệ nhân tạo triển khai tại thiết bị biên)

Chỉ số đánh giá mô hình

NSE	Nash–Sutcliffe Efficiency (Hiệu suất Nash–Sutcliffe)
RMSE	Root Mean Square Error (Sai số trung bình bình phương căn bậc hai)
MAE	Mean Absolute Error (Sai số tuyệt đối trung bình)
KGE	Kling–Gupta Efficiency (Hiệu suất Kling–Gupta)
FHV	Fraction of High Flow Volume (Tỷ lệ thể tích dòng chảy lớn)
FLV	Fraction of Low Flow Volume (Tỷ lệ thể tích dòng chảy thấp)
Peak-MAPE	Peak Mean Absolute Percentage Error (Sai số phần trăm tuyệt đối trung bình tại đỉnh lũ)
Pearson-r	Pearson Correlation Coefficient (Hệ số tương quan Pearson)
Alpha-NSE	Alpha component in NSE decomposition, biểu thị độ phân tán (do Gupta et al., 2009)
Beta-KGE	Thành phần "beta" trong phân rã KGE, phản ánh sự sai lệch trung bình giữa giá trị mô phỏng và quan sát
Beta-NSE	Beta component in NSE decomposition, biểu thị sai lệch trung bình (do Gupta et al., 2009)
Peak-Timing	Sai số về thời điểm xảy ra đỉnh lũ (lệch pha đỉnh)
Missed-Peaks	Số lượng đỉnh lũ bị mô hình bỏ sót
Z-score	Standard score (Điểm chuẩn hóa, phản ánh độ lệch chuẩn so với trung bình, thường dùng trong chuẩn hóa dữ liệu)
Min - Max	Min-Max Normalization (Chuẩn hóa dữ liệu về khoảng [0, 1] dựa trên giá trị nhỏ nhất và lớn nhất)

MỞ ĐẦU

1. Đặt vấn đề

Lũ lụt là một trong những thiên tai xuất hiện từ rất sớm và gây ảnh hưởng nghiêm trọng đến đời sống con người. Đặc biệt, trong những thập kỷ gần đây, tình trạng lũ lụt ngày càng trở nên khốc liệt và khó lường. Số liệu thống kê cho thấy số người bị ảnh hưởng bởi lũ đã tăng từ 147 triệu người/năm (1981–1990) lên 211 triệu người/năm (1991–2000) [1]. Nhiều quốc gia như Trung Quốc (2007), Thái Lan (2011), Nhật Bản (2011), Ấn Độ (2012), Mỹ (2009) hay các nước châu Âu, châu Phi cũng đã chịu thiệt hại nặng nề do lũ lụt [2]. Trong hơn 27 năm qua, lũ lụt đã khiến hơn 175.000 người thiệt mạng và gây thiệt hại kinh tế lên tới 2,2 tỷ USD toàn cầu [3]. Theo Tổ chức Khí tượng Thế giới (WMO, 2021), giai đoạn 1970–2019 ghi nhận 11.072 báo cáo thảm họa liên quan đến lũ, 2.064.929 người tử vong và tổng thiệt hại kinh tế lên tới 3,6 nghìn tỷ USD [4]. Như vậy, có thể thấy lũ lụt mang tính chất toàn cầu và diễn biến ngày càng phức tạp, gây thiệt hại nặng nề. Về quy mô, nó không dữ dội như sóng thần, bão nhưng tính ảnh hưởng của nó có tính chất kéo dài và để lại nhiều di chứng xấu cho thế hệ mai sau.

Tại Việt Nam, lũ lụt là hiện tượng phổ biến, đặc biệt nghiêm trọng ở miền Trung - khu vực có địa hình dốc, sông ngòi ngắn và lưu vực nhỏ, khiến lũ lên nhanh và khó kiểm soát. Tình trạng “lũ chồng lũ” do mưa lớn và điều tiết hồ chứa chưa hợp lý gây khó khăn lớn cho công tác phòng chống [5]. Thống kê cho thấy, chỉ riêng trong giai đoạn 1990–2010, tổng thiệt hại do lũ tại lưu vực sông Lam là khoảng 3.300 tỷ đồng; tại Vu Gia–Thu Bồn từ 1997–2009 là hơn 18.000 tỷ đồng với 765 người chết, 63 mất tích và 2.403 người bị thương; lưu vực Vệ–Trà Khúc (1996–2010) ghi nhận 601 người chết, hơn 8.500 nhà sập trôi [6].

Việc ứng phó với lũ lụt rất quan trọng, đặc biệt tại các nước đang phát triển, khi các biện pháp phòng ngừa và giảm thiểu thiên tai còn hạn chế và các đồng bằng nơi thường phải hứng chịu lũ lụt thường tập trung đông dân cư [7]. Do đó việc dự báo lũ, mực nước và lưu lượng trên sông, đặc biệt trên các sông chưa có hoặc có ít trạm quan trắc thủy văn, là rất quan trọng. Một trong những giải pháp phi công trình hữu hiệu là tạo lập cơ sở dữ liệu – thông tin về nguy cơ và mức độ ảnh hưởng của lũ lụt, giúp chính quyền và người dân chủ động ứng phó.

Tuy nhiên, dù đạt độ bao phủ toàn cầu, các mô hình hiện tại vẫn gặp hạn chế khi áp dụng tại những khu vực có ít dữ liệu đầu vào – như nhiều vùng ở Việt Nam, đặc biệt là tại thành phố Đà Nẵng. Tình trạng thiếu dữ liệu, cùng với đặc thù địa hình và điều kiện khí hậu phức tạp, khiến việc áp dụng trực tiếp các mô hình quốc tế không đạt hiệu quả như mong đợi. Do đó, việc nghiên cứu và xây dựng các mô hình dự báo lũ phù hợp với điều kiện địa phương, tận dụng hiệu quả dữ liệu sẵn có tại Việt Nam, là hết sức cần thiết.

2. Mục tiêu đề tài

Trong bối cảnh đó, việc khai thác tiềm năng của công nghệ Edge AI – cho phép xử lý và ra quyết định trực tiếp tại thiết bị biên mà không phụ thuộc hoàn toàn vào hệ thống trung tâm – đang mở ra một hướng tiếp cận đầy hứa hẹn. Edge AI không chỉ giúp giảm độ trễ trong dự báo mà còn có thể hoạt động hiệu quả tại những khu vực thiếu kết nối ổn định hoặc hạn chế về tài nguyên tính toán.

Vì vậy, việc đánh giá **tính khả thi của Edge AI trong bài toán cảnh báo lũ lụt tại khu vực Đà Nẵng** sẽ là một bước đi quan trọng, nhằm hướng đến các hệ thống cảnh báo thông minh, hiệu quả và phù hợp hơn với điều kiện thực tế tại Việt Nam.

3. Phạm vi và đối tượng nghiên cứu

Đối tượng nghiên cứu của đề tài là **ứng dụng công nghệ học máy và Edge AI vào hệ thống dự báo, cảnh báo lũ lụt**, tập trung tại **khu vực thành phố Đà Nẵng** – nơi có đặc điểm địa hình phức tạp, hạ tầng dữ liệu còn hạn chế và thường xuyên chịu ảnh hưởng bởi mưa lũ.

Phạm vi nghiên cứu bao gồm:

- Phân tích dữ liệu mưa, mực nước, địa hình tại khu vực Đà Nẵng.
- Thiết kế, thử nghiệm mô hình dự báo lũ sử dụng machine learning.
- Đề xuất giải pháp ứng dụng mô hình trên nền tảng Edge AI phù hợp điều kiện thực tế.

4. Phương pháp nghiên cứu

- Thu thập và xử lý dữ liệu: dữ liệu đầu vào gồm dữ liệu lượng mưa, mực nước, nhiệt độ, bức xạ mặt trời, độ ẩm không khí
- Phân tích và lựa chọn mô hình học máy: Đánh giá các mô hình phù hợp (LSTM, GRU, XgBoost, MTS - LSTM ,....) trong bài toán dự báo
- Huấn luyện và kiểm thử mô hình: sử dụng dữ liệu thực tế từ khu vực Đà Nẵng để huấn luyện và đánh giá mô hình

- Xây dựng và kiểm thử mô hình: Đề xuất giải pháp triển khai mô hình trên các thiết bị biên (edge device), đánh giá độ trễ, hiệu quả và khả năng mở rộng

5. Cấu trúc đồ án tốt nghiệp

Để giải quyết bài toán đặt ra là thiết kế và đánh giá tính khả thi của việc ứng dụng Edge AI trong hệ thống dự báo và cảnh báo lũ lụt tại khu vực Đà Nẵng, nhóm nghiên cứu đã xây dựng cấu trúc báo cáo gồm bốn chương chính:

- **Chương 1: Tổng quan đề tài**
- **Chương 2: Cơ sở khoa học**
- **Chương 3: Edge AI – Hướng tiếp cận mới trong dự báo và cảnh báo lũ lụt**
- **Chương 4: Kết quả huấn luyện và kiểm tra mô hình MTS–LSTM**

Chương 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1. Lý do chọn đề tài

Thành phố Đà Nẵng thuộc vùng duyên hải Nam Trung Bộ Việt Nam, là đơn vị hành chính trực thuộc Trung ương và là một trong năm thành phố lớn của cả nước. Với diện tích khoảng 1.285 km², Đà Nẵng nằm trong khu vực chuyển tiếp giữa dãy Trường Sơn và vùng đồng bằng ven biển, tiếp giáp tỉnh Thừa Thiên Huế ở phía Bắc, tỉnh Quảng Nam ở phía Nam và phía Tây, còn phía Đông giáp biển Đông. Đây là vị trí chiến lược cả về địa lý, kinh tế, quốc phòng và du lịch – một trung tâm kết nối giữa miền Bắc, miền Nam và khu vực Tây Nguyên. Về đặc điểm tự nhiên, Đà Nẵng có địa hình phức tạp, bao gồm cả khu vực đồi núi, đồng bằng và hệ thống sông ngòi dày đặc như sông Hàn, sông Cu Đê, sông Túy Loan, sông Cẩm Lệ... . Chính vì vậy, cũng làm tăng nguy cơ ngập úng vào mùa mưa khi lượng nước đổ về lớn, trong khi khả năng tiêu thoát nước còn hạn chế.

Khí hậu Đà Nẵng thuộc loại nhiệt đới gió mùa, chia thành hai mùa rõ rệt: mùa mưa (từ tháng 9 đến tháng 12) và mùa khô (từ tháng 1 đến tháng 8). Mùa mưa thường xuất hiện nhiều đợt áp thấp nhiệt đới, bão, mưa lớn kéo dài, lượng mưa trung bình hàng năm khoảng 2.500 – 3.000 mm. Đây là một trong những yếu tố chính góp phần gây ra hiện tượng ngập lụt cục bộ hoặc diện rộng tại nhiều khu vực nội đô.

Trong quá trình đô thị hóa nhanh chóng vài thập kỷ qua, diện tích bề mặt không thấm nước (như bê tông, nhựa đường) tăng lên đáng kể, làm giảm khả năng thấm hút tự nhiên của đất. Bên cạnh đó, hệ thống thoát nước mưa ở một số khu vực không được đầu tư đồng bộ hoặc đã xuống cấp theo thời gian, khiến khả năng tiêu thoát nước bị hạn chế, đặc biệt trong những tình huống có mưa lớn kéo dài hoặc mưa cực đoan trong thời gian ngắn.

Ngập lụt tại Đà Nẵng có thể chia thành ba dạng chính:

- Ngập cục bộ do mưa lớn: xảy ra trong thời gian ngắn nhưng với lượng mưa rất lớn, thường gây ngập tại các nút giao thông, tuyến đường trũng hoặc thấp hơn mặt bằng chung.
- Ngập do lũ từ thượng nguồn: nước từ các con sông chảy về, kết hợp với mưa lớn gây tràn bờ, ảnh hưởng đến các khu dân cư ven sông.
- Ngập do triều cường hoặc nước biển dâng: xuất hiện ở các khu vực gần biển như Sơn Trà, Ngũ Hành Sơn vào những thời điểm triều lên cao.

Đặc biệt, trong những năm gần đây, ảnh hưởng của biến đổi khí hậu ngày càng rõ nét, biểu hiện qua các hình thái thời tiết cực đoan, mưa lớn bất thường, thời gian mưa kéo

dài hoặc tập trung vào một thời điểm ngắn. Điều này làm tăng tính bất định của hiện tượng ngập lụt và gây khó khăn trong công tác dự báo, phòng chống và ứng phó. Điển hình như, năm 2022, Đà Nẵng phải đối mặt với những đợt mưa lớn xuất hiện liên tục trong các tháng 9 và 10, điển hình là đợt mưa kéo dài trong nhiều giờ vào ngày 14/10/2022 đã khiến lượng mưa đo được tại một số điểm lên đến hơn 500cm, vượt xa mức trung bình nhiều năm. Tình trạng mưa lớn kéo dài đã dẫn đến ngập lụt trên diện rộng, đặc biệt tại các tuyến đường trung tâm như Điện Biên Phủ, Lê Duẩn, Nguyễn Hữu Thọ, Trường Chinh, và các khu vực trũng thấp như quận Thanh Khê, Cẩm Lệ, Hải Châu. Nhiều khu dân cư bị cô lập, hệ thống giao thông tê liệt tạm thời và các hoạt động sinh hoạt của người dân bị ảnh hưởng nặng nề. Bên cạnh thiệt hại vật chất, đợt mưa lũ này cũng đặt ra thách thức lớn về công tác điều hành, phối hợp ứng cứu và năng lực dự báo, cảnh báo tại chỗ.

Sang năm 2023, mặc dù lượng mưa trung bình toàn thành phố có xu hướng giảm nhẹ so với năm trước, tuy nhiên các hiện tượng mưa lớn cục bộ vẫn xuất hiện và gây ra những ảnh hưởng không nhỏ. Đặc biệt trong tháng 10/2023, có thời điểm lượng mưa đạt từ 200 đến 300 cm chỉ trong vòng 6 giờ, gây ngập úng tại một số khu vực quen thuộc như Trường Chinh, Hàm Nghi, Nguyễn Văn Linh và Hòa Khánh. Ngập lụt trong năm 2023 không diễn ra liên tục như năm trước, nhưng lại mang tính chất bất ngờ, khó dự đoán và xuất hiện dưới dạng "ngập cục bộ" với tốc độ nước rút chậm, làm cản trở sinh hoạt và giao thông. Một số hệ thống trạm đo tự động đã phát huy hiệu quả trong việc theo dõi lượng mưa, mực nước, nhờ đó cơ quan chức năng có thể đưa ra cảnh báo sớm và điều tiết giao thông linh hoạt hơn.

Trước thực trạng trên, việc nâng cao năng lực dự báo, cảnh báo ngập lụt là nhu cầu cấp thiết. Tuy nhiên, nhiều hệ thống dự báo hiện nay vẫn phụ thuộc vào các mô hình tập trung, yêu cầu hạ tầng tính toán lớn và chậm trong phản hồi. Trong khi đó, công nghệ mới như **Edge AI** – trí tuệ nhân tạo triển khai tại các thiết bị biên – có tiềm năng cung cấp các giải pháp cảnh báo sớm, xử lý dữ liệu tại chỗ, giảm độ trễ và tối ưu chi phí triển khai.

Vì vậy, nhóm nghiên cứu lựa chọn đề tài này nhằm giải quyết các vấn đề sau:

- Làm rõ hiện trạng ngập lụt tại Đà Nẵng dưới góc nhìn dữ liệu và công nghệ;
- Tìm hiểu khả năng ứng dụng các kỹ thuật trí tuệ nhân tạo hiện đại trong dự báo, cảnh báo lũ lụt;

- Đề xuất mô hình ứng dụng **Edge AI** trong việc xây dựng hệ thống cảnh báo ngập lụt thời gian thực phù hợp với điều kiện hạ tầng của thành phố;
- Góp phần nâng cao năng lực ứng phó thiên tai, giảm thiểu thiệt hại và đảm bảo an toàn cho người dân trong bối cảnh đô thị hóa và biến đổi khí hậu.

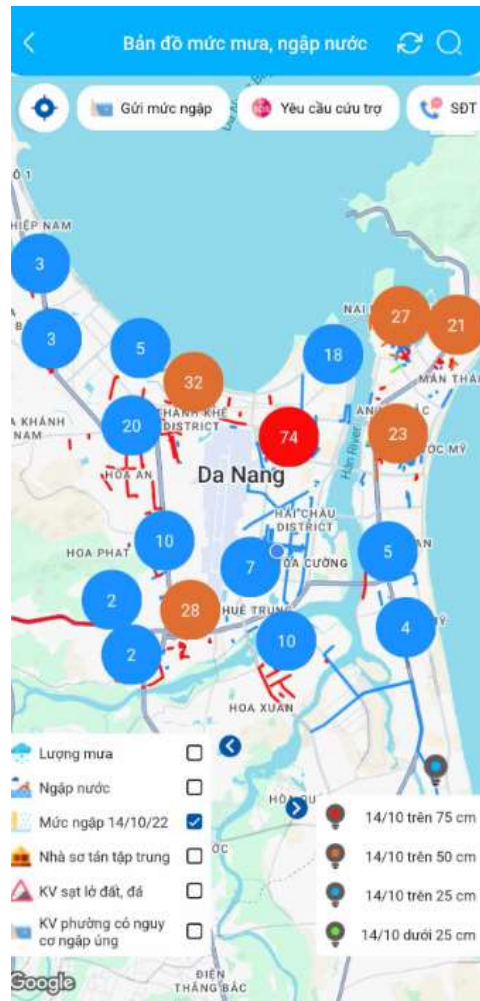
1.2. Hệ thống quan trắc ngập lụt tại Đà Nẵng

1.2.1. Hiện trạng và giải pháp lũ lụt hiện nay tại thành phố Đà Nẵng

Lũ lớn tại Đà Nẵng thường xảy ra vào mùa mưa (tháng 9–11), do ảnh hưởng của bão, áp thấp nhiệt đới và không khí lạnh kết hợp gió mùa Đông Bắc. Với vị trí địa lý đặc thù, thành phố thường hứng chịu mưa lớn kéo dài, nhất là khi kết hợp với lũ từ hệ thống sông Vu Gia – Thu Bồn đổ về qua các sông nhánh như Cẩm Lệ.

Thống kê cho thấy các đợt lũ lớn năm 2009, 2013 và đặc biệt là 2020 đã khiến mực nước tại trạm Cẩm Lệ vượt báo động III (trên 3,4m), gây ngập nặng tại Cẩm Lệ, Hòa Vang, Hải Châu và trung tâm thành phố [24]. Riêng đợt lũ tháng 10/2020 ghi nhận lượng mưa trên 700cm/24 giờ, làm hơn 10.000 hộ dân bị ảnh hưởng và gây thiệt hại nghiêm trọng về hạ tầng [25].

Hiện nay, một điểm thuận lợi đáng kể là thành phố Đà Nẵng hiện đã có mạng lưới các trạm đo khí tượng thủy văn tương đối dày đặc được lắp đặt trên nhiều khu vực trọng yếu. Đây là cơ sở quan trọng cho việc xây dựng hệ thống theo dõi, dự báo và cảnh báo ngập lụt theo thời gian thực. Các trạm này chủ yếu được quản lý bởi Sở Tài nguyên và Môi trường, Trung tâm Quan trắc tài nguyên môi trường thành phố, cũng như hợp tác với các đơn vị công nghệ như Watec



Hình 1.1. Bản đồ mức mưa, ngập nước tại thành phố Đà Nẵng (Nguồn: app Smart City)

Hiện nay, Đà Nẵng đã xây dựng mạng lưới trạm đo khí tượng thủy văn tương đối dày đặc tại các khu vực trọng điểm, do Sở Tài nguyên và Môi trường cùng Trung tâm Quan trắc tài nguyên môi trường quản lý, phối hợp với các đơn vị công nghệ như Watec. Dữ liệu từ các trạm này được tích hợp trên nhiều nền tảng, bao gồm ứng dụng Danang Smart City của thành phố và ứng dụng riêng của Watec, giúp cơ quan chức năng và người dân theo dõi tình hình mưa lũ theo thời gian thực và kịp thời đưa ra cảnh báo. Tuy nhiên, hệ thống này chỉ giúp người dân theo dõi tình hình mưa lũ, và chưa có khả năng dự báo lũ lụt để giúp cho người dân có thể biết và lên kế hoạch để phòng tránh những thiệt hại cho tài sản và tinh thần người dân

1.2.2. Phân cứng của các thiết bị điều khiển chính giám sát thu thập dữ liệu

Thành phố Đà Nẵng đã chủ động triển khai hệ thống các trạm quan trắc thông minh tại nhiều vị trí trọng điểm trên địa bàn nhằm giám sát tình hình mưa và mực nước

một cách liên tục, chính xác. Những trạm này được bố trí tại các khu vực trọng điểm góp phần xây dựng cơ sở dữ liệu thực tế để phục vụ công tác dự báo và cảnh báo ngập lụt. Các trạm đo chủ yếu bao gồm: trạm đo lượng mưa và trạm đo mực nước, sử dụng các công nghệ cảm biến hiện đại như cảm biến tipping bucket, cảm biến siêu âm và radar. Dữ liệu được thu thập theo thời gian thực và truyền về trung tâm điều hành thông qua các phương thức kết nối không dây như LoRa, 3G/4G, tùy theo hạ tầng khu vực. Ngoài ra, việc tích hợp các trạm đo này vào ứng dụng Danang Smart City cho phép người dân cũng như các cơ quan chức năng theo dõi, phân tích, và phản ứng kịp thời trước các diễn biến bất thường của thời tiết và mực nước.

a) Tổng quan về trạm đo mưa

Trạm đo mưa tự động Vrain là một thiết bị hiện đại được thiết kế để phục vụ công tác quan trắc lượng mưa trong thời gian thực, hỗ trợ hiệu quả cho các hoạt động phòng chống thiên tai. Hệ thống này nổi bật với khả năng tự động hóa cao, tích hợp nhiều công nghệ hiện đại và được triển khai phổ biến tại nhiều địa phương trong cả nước.



Hình 1.2. Trạm đo mưa tự động Watec

Dữ liệu từ cảm biến chao lật được xử lý bởi bộ vi điều khiển trung tâm tích hợp trong trạm, sử dụng dòng vi điều khiển ARM Cortex-M3 32-bit với tốc độ 32 MHz, cho phép thu thập, xử lý và lưu trữ số liệu một cách hiệu quả. Dữ liệu lượng mưa được lưu vào thẻ nhớ microSD (dung lượng lên đến 32 GB) và truyền về trung tâm thông qua các module truyền thông như GSM, GPRS, 3G hoặc 4G, đảm bảo khả năng kết nối linh hoạt ngay cả ở những vùng sâu vùng xa.

Trạm hoạt động hoàn toàn độc lập nhờ sử dụng nguồn năng lượng mặt trời đi kèm pin sạc Lithium-ion 3.7V DC dung lượng 6.8Ah, đủ để duy trì hoạt động ổn định trong khoảng 45 đến 60 ngày mà không cần sạc lại. Tần suất truyền dữ liệu có thể được cấu hình linh hoạt, ví dụ như mỗi 10 phút/lần hoặc 60 phút/lần. Hệ thống cũng cho phép người dùng điều khiển và cấu hình từ xa qua tin nhắn SMS hoặc giao diện web, đồng thời hỗ trợ cập nhật phần mềm firmware từ xa thông qua mạng di động.

b) Tổng quan về trạm đo mực nước tự động



Hình 1.3. Trạm đo mực nước tự động Watec

Trạm đo mực nước tự động sử dụng công nghệ Radar là một hệ thống hiện đại được thiết kế để thực hiện việc quan trắc mực nước tại các khu vực như sông, ao, hồ,... trong thời gian thực. Hệ thống này được ứng dụng rộng rãi trong các dự án thủy lợi, phòng chống thiên tai, cảnh báo lũ lụt. Toàn bộ trạm được xây dựng từ các thiết bị có độ chính xác cao, hoạt động bền bỉ trong điều kiện khắc nghiệt, và đặc biệt có khả năng tự động hóa cao trong việc thu thập, xử lý và truyền tải dữ liệu.

1.3. Các nghiên cứu trước đó

Nhiều dự án cũng như nghiên cứu của các nhà khoa học trong và ngoài nước đã ứng dụng các mô hình dựa trên tính chất vật lý (physically-based models) có độ chính xác cao vào việc dự đoán mực nước, lưu lượng hay dòng chảy trên các con sông, hồ chứa như: MIKE, HYDRO River, HEC-HMS, SOBEK, EFDC... Tuy nhiên, các mô hình này cần rất nhiều thông số đầu vào như mực nước, lưu lượng, độ ẩm đất, bốc hơi, tốc độ

thấm, và đòi hỏi thời gian mô phỏng dài cùng sự tham gia của các chuyên gia thủy văn – thủy lực. Điều này làm giảm tính khả thi khi triển khai trong thời gian thực.

Hướng phát triển trong tương lai của thủy văn và quản lý tài nguyên nước là tích hợp các mô hình toán truyền thống với mô hình học máy (machine learning) để xử lý và phân tích trực tiếp từ dữ liệu lớn [8]. Trong những năm gần đây, học máy đã được ứng dụng rộng rãi nhờ khả năng quản lý và khai thác dữ liệu tốt. Các nghiên cứu tổng quan đã chỉ ra học máy ngày càng được sử dụng trong quản lý tài nguyên nước và rủi ro thiên tai.

Các mô hình định hướng dữ liệu (data-driven models) sử dụng học máy đã được so sánh với các mô hình vật lý về mô phỏng lũ, dòng chảy mặt, thoát nước, bùn cát, mực nước ngầm... và cho thấy hiệu quả đáng kể [9,10]. Một số nghiên cứu dùng học sâu (deep learning) để xây dựng bản đồ ngập [11], tuy nhiên trước ảnh hưởng của biến đổi khí hậu, việc dự đoán mực nước, lượng mưa và dòng chảy trở nên phổ biến hơn [12–16].

Một số nghiên cứu tiêu biểu như:

- **Hệ thống Delft-FEWS (Deltares, Hà Lan):** dùng mô hình thủy lực và dữ liệu khí tượng để dự báo chi tiết mực nước và dòng chảy, ứng dụng ở nhiều quốc gia [17].
- **Nghiên cứu “Climate change is making rain difficult to predict...”:** sử dụng mạng ANN để dự báo lượng mưa trong điều kiện thời tiết biến động, ứng dụng trong nông nghiệp và thiên tai [19].
- **Nhiều bài báo đã so sánh hiệu quả của các thuật toán ML như:** hồi quy tuyến tính, SVM, XGBoost, LGBMR, GPR... để đánh giá hiệu suất dự đoán mực nước theo giờ, theo ngày [20–23].

Ngoài ra, các công ty công nghệ như **Google, OpenWeather, Open-Meteo** cũng phát triển các mô hình AI dự báo mưa miễn phí, có độ chính xác cao. Trong đó, mô hình **LSTM** kết hợp nhiều nguồn dữ liệu (ECMWF, ERA5-Land, NASA IMERG, HydroATLAS...) đã cho phép dự báo trước 5 ngày với độ chính xác tương đương hoặc vượt hệ thống GloFAS – đặc biệt hiệu quả ở các lưu vực nhỏ, thiếu dữ liệu [18].

1.4. Kết luận chương

Từ những phân tích của chương 1, có thể thấy ngập lụt tại Đà Nẵng ngày càng phức tạp và khó lường do tác động của biến đổi khí hậu, đô thị hóa nhanh và hạ tầng

thoát nước chưa đáp ứng yêu cầu. Với vị trí địa lý đặc thù gồm cả đồi núi, đồng bằng và vùng ven biển, cùng hệ thống sông ngòi dày đặc, Đà Nẵng thường xuyên hứng chịu các loại hình ngập: mưa lớn cục bộ, lũ từ thượng nguồn và triều cường. Đặc biệt, các năm gần đây (2022, 2023) ghi nhận nhiều đợt mưa cực đoan kéo dài, gây thiệt hại lớn về kinh tế và đời sống.

Dù thành phố đã triển khai mạng lưới trạm quan trắc tự động với cảm biến hiện đại và truyền dữ liệu thời gian thực (qua ứng dụng Danang Smart City), hệ thống cảnh báo ngập hiện tại vẫn chủ yếu mô tả hiện trạng, thiếu khả năng dự báo sớm chính xác. Điều này cho thấy cần tích hợp công nghệ mới như trí tuệ nhân tạo, học máy và mô hình thủy văn để nâng cao hiệu quả dự báo. Việc phát triển một hệ thống dự báo lũ hiện đại, kết hợp dữ liệu quan trắc thời gian thực và tính toán thông minh là cấp thiết, nhằm hỗ trợ công tác phòng chống thiên tai, tăng cường năng lực quản lý rủi ro, và hướng tới mục tiêu xây dựng thành phố thông minh, thích ứng với biến đổi khí hậu.

Chương 2: CƠ SỞ KHOA HỌC

2.1. Dữ liệu Khí tượng – Thủy văn tại Đà Nẵng

Trong nghiên cứu này, chúng tôi sử dụng **dữ liệu quan trắc thực tế** về mưa và mực nước tại các trạm khí tượng – thủy văn trên địa bàn thành phố Đà Nẵng (Việt Nam). Bộ dữ liệu bao gồm các chuỗi thời gian ghi nhận **lượng mưa (cm)** và **mực nước sông (m)** theo thời gian, được thu thập với tần suất cố định (theo giờ hoặc ngày) từ mạng lưới trạm đo địa phương. Mỗi bản ghi dữ liệu gồm các trường: thời gian quan trắc, lượng mưa đo được tại trạm mưa, và mực nước tương ứng tại trạm thủy văn.

Cấu trúc và nguồn gốc dữ liệu: Dữ liệu mưa và mực nước đến từ các trạm quan trắc do Đài Khí tượng Thủy văn cung cấp, đảm bảo độ tin cậy và tính liên tục. Lượng mưa được đo bằng cảm biến mưa tự động, ghi nhận cường độ và thời gian mưa – hai yếu tố quan trọng quyết định nguy cơ lũ lụt [26]. Mực nước được đo tại các trạm thủy văn dọc sông Hàn và các sông suối chính, phản ánh mức nước dâng do nước mưa đổ về. Các cảm biến mực nước ghi nhận mực nước tại chỗ liên tục, cho phép theo dõi dòng chảy và mức độ ngập trong thời gian thực [26]. Bộ dữ liệu được thu thập liên tục trong nhiều năm, bao quát cả giai đoạn mùa khô và mùa mưa, bao gồm cả những sự kiện mưa lũ lớn bất thường.

Đặc điểm khí hậu địa phương: Đà Nẵng nằm ở vùng duyên hải miền Trung, có mùa mưa từ khoảng tháng 9 đến tháng 12 hàng năm. Lượng mưa trung bình năm vào khoảng 2.000–2.500 cm, tập trung chủ yếu vào các tháng mùa mưa, với đỉnh điểm thường rơi vào tháng 10 và 11 [27, 28]. Vào mùa khô (khoảng tháng 4–8), lượng mưa rất thấp; ngược lại, mùa mưa có thể xuất hiện những trận mưa lớn kéo dài nhiều giờ hoặc nhiều ngày liên tiếp. Thống kê lịch sử cho thấy tần suất và cường độ mưa cực đoan tại Đà Nẵng đang có xu hướng gia tăng do tác động của bão nhiệt đới và gió mùa đông lạnh. Trước năm 1998, lũ quét lớn chỉ xảy ra khoảng một lần mỗi thập kỷ, nhưng giai đoạn 1998–2009 đã ghi nhận 6 trận lũ quét, tất cả đều vượt mức báo động cấp 3 (mức cao nhất) [29]. Đặc biệt, trận lũ lịch sử tháng 10/2022 do bão Son Ca đã gây mưa tới 795 cm chỉ trong một đêm, khiến toàn thành phố ngập sâu 1,5 m và gây thiệt hại nặng nề [27]. Những sự kiện cực đoan này thể hiện biên độ dao động rất lớn của dữ liệu, từ những ngày không mưa hoặc mưa nhỏ đến những ngày mưa kỷ lục hàng trăm cm.

Phân tích các đặc trưng dữ liệu: Lượng mưa có phân bố rất lệch – phần lớn thời gian là mưa nhỏ hoặc không mưa, đan xen với một số ít sự kiện mưa rất lớn. Chuỗi mưa xuất hiện nhiều giá trị 0 hoặc gần 0, trong khi các giá trị cực đại có thể lên đến vài trăm

cm/ngày (ví dụ trận mưa ~795 cm đêm 14/10/2022) [27]. Ngược lại, chuỗi mực nước sông biến động tương đối liên tục hơn, phản ánh hiệu ứng tích lũy của nước mưa trong lưu vực. Mực nước thường tăng chậm và đạt đỉnh sau các trận mưa lớn (do độ trễ lưu vực), rồi giảm dần sau khi mưa tạnh. Sự tương quan thời gian giữa mưa và mực nước là một đặc trưng quan trọng: thông thường, mực nước dâng cao nhất sau đỉnh mưa vài giờ đến vài chục giờ, tùy thuộc vào kích thước và điều kiện lưu vực. Nhìn chung, những ngày có mưa lớn thường đi kèm với mực nước dâng cao, và ngược lại những giai đoạn ít mưa thì mực nước ở mức cơ bản.

Để hiểu rõ hơn mối liên hệ giữa các biến, chúng tôi tiến hành phân tích thống kê sơ bộ. Hệ số tương quan giữa lượng mưa ngày và mực nước ngày liền sau cho thấy giá trị dương khá cao, hàm ý rằng mưa lớn thường kéo theo mực nước cao. Thống kê cho thấy mưa trên ngưỡng ~100 cm/ngày hầu như luôn gây lũ ở mức báo động tại Đà Nẵng. Chẳng hạn, dữ liệu cho thấy trong đợt lũ 2020, việc kết hợp lượng mưa bình quân lưu vực và mực nước thượng nguồn cho phép dự báo mực nước tại trạm chính với trước thời gian 20 phút, hỗ trợ cảnh báo sớm lũ quét [30]. Điều này gợi ý rằng việc sử dụng kết hợp thông tin mưa và mực nước (đa biến) có thể cải thiện độ chính xác của dự báo mực nước. Ngoài ra, chúng tôi cũng xem xét các ngưỡng cảnh báo: ví dụ mực nước trên 2,5 m tương ứng báo động cấp 3 tại sông Hàn. Những đỉnh lũ lịch sử (như năm 1964, 1999) đều vượt xa ngưỡng này [29], cho thấy dữ liệu có những ngoại lệ rất lớn cần lưu ý trong quá trình mô hình hóa.

Tóm lại, dữ liệu đầu vào cho mô hình dự báo lũ bao gồm chuỗi lượng mưa và mực nước theo thời gian với sự biến động mạnh theo mùa và theo sự kiện. Việc phân tích đặc trưng cho thấy rõ sự phụ thuộc động lực: mưa lớn là tác nhân chính gây lũ, và mực nước phản ứng với độ trễ nhất định. Những hiểu biết này sẽ định hướng cách xây dựng tập dữ liệu huấn luyện cho mô hình LSTM, đảm bảo mô hình có thể học được quan hệ phi tuyến giữa mưa và mực nước trong quá khứ và tương lai.

2.1.1. Dữ liệu theo ngày (Daily):

A	B	C	D	E	F
Date	Luong_mua	Nhiet_do_cao_nhat	Nhiet_do_thap_nhat	Buc_xa_mat_troi	Do_am_khong_khi
2023-01-01 00:00:00	0	21.4	19.4	60.41628	82
2023-01-02 00:00:00	1.4	22.1	20	67.1292	87
2023-01-03 00:00:00	1	21.8	20.7	46.52748	87
2023-01-04 00:00:00	0.8	22.3	20.9	50.11542	86
2023-01-05 00:00:00	2	23.2	21.8	106.82802	84
2023-01-06 00:00:00	1.2	22.4	20.6	95.13828	79
2023-01-07 00:00:00	0	21.9	20.3	50.3469	80
2023-01-08 00:00:00	0	21.8	20.1	41.78214	85
2023-01-09 00:00:00	0.2	23.6	20.6	74.6523	87
2023-01-10 00:00:00	0	24.3	20.3	130.32324	87
2023-01-11 00:00:00	0	24.7	18.9	162.96192	89
2023-01-12 00:00:00	0	24.9	20	196.06356	91
2023-01-13 00:00:00	0	24.6	21.1	207.05886	93
2023-01-14 00:00:00	0	26.7	18.8	234.02628	93
2023-01-15 00:00:00	0	26.2	19.1	228.47076	92
2023-01-16 00:00:00	6.2	22.5	19.7	50.80986	87
2023-01-17 00:00:00	0	21.9	19.5	57.63852	85
2023-01-18 00:00:00	0.8	21.1	19.4	79.16616	84
2023-01-19 00:00:00	0	21.8	19	165.73968	80

Hình 2.1. Dữ liệu khí tượng – Thủy văn theo ngày tại thành phố Đà Nẵng

Nguồn: Tổng hợp từ các trạm quan trắc địa phương, kết hợp với dữ liệu bổ sung từ web có hỗ trợ dữ liệu nguồn mở như: Open-meteo, Windy, ...

- Thời gian: Từ ngày 01/01/2023 đến hết năm 31/12/2024, đo từng ngày.
- Các trường dữ liệu gồm:
 - Date: Ngày đo (định dạng Year – Month - Day)
 - Luong_mua: Lượng mưa tổng ngày (cm)
 - Nhiet_do_cao_nhat: Nhiệt độ tối đa trong ngày (°C)
 - Nhiet_do_thap_nhat: Nhiệt độ tối thiểu trong ngày (°C)
 - Buc_xa_mat_troi: Tổng bức xạ mặt trời trong ngày (W/m²)
 - Do_am_khong_khi: Độ ẩm trung bình trong ngày (%)

2.1.2. Dữ liệu theo giờ (Hourly):

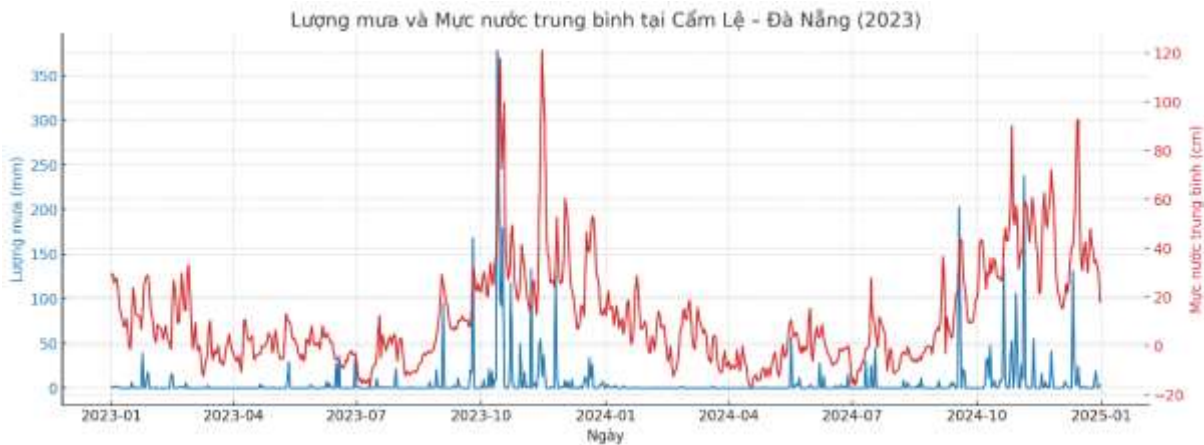
A	B	C	D	E	F	G
Tram	Datetime	Muc_nuoc	Luong_mua	Nhiet_do	Buc_xa_mat_troi	Boc_thoat_hoi_nuoc_tiem_nang
camle	2023-01-01 00:00:00	11	0	20.6	0	0.03
camle	2023-01-01 01:00:00	5	0	20.2	0	0.03
camle	2023-01-01 02:00:00	8	0	19.6	0	0.02
camle	2023-01-01 03:00:00	10	0	19.6	0	0.02
camle	2023-01-01 04:00:00	14	0	19.6	0	0.02
camle	2023-01-01 05:00:00	20	0	19.4	0	0.01
camle	2023-01-01 06:00:00	20	0	19.4	0	0.01
camle	2023-01-01 07:00:00	25	0	19.7	6	0.05
camle	2023-01-01 08:00:00	22	0	20.1	53	0.07
camle	2023-01-01 09:00:00	22	0	20.1	79	0.08
camle	2023-01-01 10:00:00	14	0	20.4	126	0.11
camle	2023-01-01 11:00:00	17	0	21.4	310	0.21
camle	2023-01-01 12:00:00	17	0	21.2	276	0.2
camle	2023-01-01 13:00:00	18	0	21	200	0.16
camle	2023-01-01 14:00:00	25	0	21	182	0.15
camle	2023-01-01 15:00:00	36	0	20.9	112	0.12

Hình 2.2. Dữ liệu khí tượng thủy văn theo giờ tại thành phố Đà Nẵng

Nguồn: Được ghi nhận tự động tại trạm Cẩm Lệ, dữ liệu liên tục theo từng giờ.

- Thời gian: Từ ngày 01/01/2023 đến hết năm 31/12/2024, theo từng khung giờ.
- Các trường dữ liệu gồm:
- Tram: Tên trạm đo (Cẩm Lệ)
- Datetime: Thời gian đo chi tiết theo giờ (định dạng Year – Month – Day Hour – Minute - Second)
- Muc_nuoc: Mực nước tại trạm (cm)
- Luong_mua: Lượng mưa trong từng giờ (cm)
- Nhiet_do: Nhiệt độ từng giờ (°C)
- Buc_xa_mat_troi: Bức xạ mặt trời (W/m²)
- Boc_thoat_hoi_tiem_nang: Bốc thoát hơi nước tiềm năng FAO (cm/h)

Kết quả phân tích thực tế từ trạm Cẩm Lệ-Đà Nẵng:



Hình 2.3. Biểu đồ thống kê Lượng mưa và Mức nước trung bình tại Cẩm Lệ - Đà Nẵng (năm 2023 và 2024)

Nhận xét

- **Phân tích tương quan: Lượng mưa và Mức nước**
 - + Biểu đồ heatmap cho thấy hệ số tương quan tuyến tính dương (~ 0.61) giữa lượng mưa và mực nước trung bình theo ngày.
 - + Điều này xác nhận giả thuyết rằng các ngày mưa lớn thường kéo theo mực nước sông dâng cao, và hai yếu tố này có mối quan hệ thống kê đáng kể – là nền tảng phù hợp để áp dụng mô hình học máy (LSTM).
- **Biến động lượng mưa và mực nước theo thời gian (2023)**
 - + Đỉnh mưa tập trung rõ rệt vào các tháng mùa mưa (tháng 9–12).
 - + Mực nước có độ trễ nhẹ so với lượng mưa, phản ánh đúng quy luật dòng chảy lưu vực.
 - + Các đỉnh lớn của mực nước thường xảy ra 1–2 ngày sau các đợt mưa lớn, cho thấy hiệu ứng tích tụ và dòng thoát chậm của địa hình Đà Nẵng.

2.2. Tiền xử lý dữ liệu

Dữ liệu thô thu thập từ thực tế thường chứa nhiều vấn đề chất lượng cần được xử lý trước khi đưa vào mô hình học máy. Trong nghiên cứu này, chúng tôi thực hiện quy trình tiền xử lý chặt chẽ nhằm làm sạch và chuẩn hóa dữ liệu, bao gồm các bước chính như sau:

- **Kiểm tra và xử lý dữ liệu thiếu:** Trước hết, tiến hành kiểm tra các giá trị khuyết thiếu (NA/null) trong chuỗi thời gian mưa và mực nước. Dữ liệu đo thực tế có thể bị thiếu do lỗi thiết bị hoặc gián đoạn truyền dẫn. Những khoảng trống dữ liệu này, nếu không xử lý, sẽ gây sai lệch cho mô hình. Chúng tôi áp dụng phương

pháp nội suy (interpolation) theo thời gian để điền giá trị cho các điểm thiếu nhỏ lẻ, đảm bảo tính liên tục của chuỗi. Cụ thể, đối với chuỗi mực nước, nếu thiếu một vài giờ, có thể nội suy tuyến tính dựa trên các giá trị lân cận; đối với chuỗi mưa, nếu thiếu dữ liệu nhưng mực nước cho thấy biến động lớn, có thể sử dụng thông tin mực nước để ước tính lượng mưa bị thiếu tương ứng. Trường hợp khoảng thiếu quá dài hoặc không thể nội suy đáng tin cậy, chúng tôi loại bỏ những đoạn dữ liệu đó khỏi tập huấn luyện. Việc xử lý cẩn thận dữ liệu thiếu là rất quan trọng, vì dữ liệu thiếu hoặc nhiễu có thể ảnh hưởng xấu đến hiệu suất của mô hình. Nghiên cứu cũng chỉ ra rằng việc bổ sung thêm các nguồn dữ liệu khác (ví dụ: độ ẩm đất, lưu lượng thượng nguồn) có thể giúp cải thiện dự báo khi dữ liệu mưa/mực nước bị thiếu, tuy nhiên trong phạm vi luận văn này chúng tôi tập trung vào hai biến chính là mưa và mực nước.

- **Phát hiện và loại bỏ ngoại lệ:** Tiếp theo, chúng tôi sàng lọc các giá trị ngoại lệ (outlier) trong dữ liệu. Ngoại lệ có thể do lỗi đo (ví dụ cảm biến hỏng cho giá trị âm hoặc quá lớn phi thực tế) hoặc các hiện tượng bất thường không đại diện cho quy luật chung. Chúng tôi áp dụng phương pháp thống kê để phát hiện ngoại lệ: sử dụng ngưỡng vật lý (chẳng hạn lượng mưa không thể âm, mực nước không thể giảm đột ngột xuống dưới 0) và ngưỡng thống kê (ví dụ giá trị vượt quá 3σ so với trung bình). Những mẫu vượt quá ngưỡng này được đánh giá thủ công; nếu được xác định là lỗi, sẽ được loại bỏ hoặc hiệu chỉnh. Ví dụ, nếu một trạm mưa ghi nhận 500 cm trong 5 phút – đây là không thực tế và sẽ bị loại. Tương tự, nếu mực nước giảm đột ngột 2 m trong 15 phút rồi tăng lại, có thể là lỗi thiết bị; chúng tôi sẽ loại bỏ đoạn dữ liệu đó. Việc làm sạch ngoại lệ đảm bảo rằng mô hình không học phải các quan hệ sai lệch từ dữ liệu nhiễu.
- **Làm mịn và đồng bộ dữ liệu:** Dữ liệu mưa và mực nước được đồng bộ về cùng một mốc thời gian. Nếu tần suất đo của các trạm khác nhau (ví dụ mưa đo theo giờ, mực nước theo 15 phút), chúng tôi lấy khoảng thời gian chung nhỏ nhất (15 phút) và nội suy hoặc lấy trung bình cho chuỗi mưa theo cùng bước thời gian. Đồng thời, có thể áp dụng làm mịn (smoothing) nhẹ cho mực nước để giảm ảnh hưởng của dao động nhiễu tần số cao, bằng cách sử dụng trung bình trượt ngắn hạn. Tuy nhiên, việc làm mịn được thực hiện cẩn trọng để không làm mất tín hiệu đỉnh lũ quan trọng.

2.3. Chuẩn hóa dữ liệu (Normalization)

Sau khi dữ liệu đã sạch và đồng bộ, toàn bộ các đặc trưng (biến đầu vào) đều được chuẩn hóa về cùng một quy mô nhằm giúp quá trình huấn luyện mạng LSTM ổn định hơn. Chúng tôi cân nhắc hai phương pháp chuẩn hóa thông dụng là Min-Max scaling và Z-score:

- **Chuẩn hóa Min-Max:** Đưa giá trị mỗi biến về khoảng $[0, 1]$ dựa trên giá trị nhỏ nhất và lớn nhất của biến đó.

+ Công thức:
$$X' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (2.1)$$

- + Phương pháp này đảm bảo mọi giá trị mưa và mực nước đều nằm trong $[0,1]$, giúp các hệ số mạng học tập không bị chi phối bởi độ lớn tuyệt đối của dữ liệu. Tuy nhiên, Min-Max khá nhạy cảm với ngoại lệ: một giá trị cực lớn có thể thu hẹp khoảng biến thiên của phần lớn dữ liệu còn lại.
- **Chuẩn hóa Z-score:** Chuẩn hóa dựa trên độ lệch chuẩn, đưa dữ liệu về trung bình 0 và độ lệch chuẩn 1.

+ Công thức:
$$x'' = \frac{x - \mu}{\sigma} \quad (2.2)$$

- + trong đó μ là giá trị trung bình và σ là độ lệch chuẩn của biến.
- + Z-score ít bị ảnh hưởng bởi ngoại lệ hơn, nhưng kết quả chuẩn hóa không bị chặn trong một khoản

Trong nghiên cứu này, chúng tôi thử nghiệm cả hai phương pháp và nhận thấy chuẩn hóa Min-Max mang lại kết quả tốt hơn cho mô hình LSTM trên tập dữ liệu này. Điều này phù hợp với một số nghiên cứu gần đây, khi so sánh Min-Max và Z-score cho LSTM đã chỉ ra rằng Min-Max thường cho sai số thấp hơn và độ chính xác dự báo cao hơn trong nhiều trường hợp [32]. Nguyên nhân có thể do LSTM sử dụng hàm kích hoạt sigmoid và tanh, vốn hoạt động hiệu quả khi dữ liệu nằm trong khoảng $[-1,1]$ hoặc $[0,1]$. Vì vậy, toàn bộ các đầu vào (lượng mưa, mực nước) được chuẩn hóa về $[0,1]$ dựa trên giá trị min-max của từng biến trong tập huấn luyện. Việc chuẩn hóa được thực hiện sau khi tách tập dữ liệu, để tránh rò rỉ thông tin từ tập kiểm tra.

- **Tạo tập dữ liệu huấn luyện mô hình:** Sau khi chuẩn hóa, chúng tôi tiến hành tạo các mẫu huấn luyện cho mô hình LSTM từ chuỗi thời gian đa biến. Do LSTM là mô hình dự báo chuỗi thời gian, ta cần chuyển đổi dữ liệu chuỗi đơn biến (univariate) hoặc đa biến (multivariate) thành các cặp đầu vào – đầu ra phù hợp.

Cách làm thông dụng là sử dụng cửa sổ trượt thời gian (sliding window). Cụ thể, chúng tôi xác định một độ dài cửa sổ L (số bước thời gian quá khứ đưa vào mô hình) dựa trên khả năng “nhớ” cần thiết. Với bài toán dự báo lũ, L có thể là vài giờ đến vài ngày tùy theo độ trễ lưu vực. Mỗi mẫu huấn luyện sẽ gồm đầu vào là một chuỗi liên tiếp L bước thời gian của (mưa, mực nước) và đầu ra là giá trị mực nước tại bước thời gian kế tiếp (hoặc vài bước kế tiếp nếu dự báo nhiều bước). Ví dụ, nếu $L=24$ giờ, một mẫu có thể là dữ liệu mưa và mực nước từ 0h ngày 1 đến 23h ngày 1, dùng để dự đoán mực nước tại 0h ngày 2. Quá trình trượt cửa sổ qua toàn bộ chuỗi sẽ tạo ra hàng nghìn mẫu, đảm bảo bao quát đủ các tình huống (tăng, giảm của mực nước) để mô hình học.

- **Phân chia tập huấn luyện và kiểm tra:** Cuối cùng, dữ liệu được chia thành tập huấn luyện và tập kiểm tra để đánh giá mô hình. Thông thường chúng tôi sử dụng tỷ lệ 80/20, tức ~80% dữ liệu cho huấn luyện, ~20% còn lại để kiểm chứng kết quả dự báo. Việc chia theo thời gian được thực hiện cẩn thận để không làm gián đoạn tính liên tục của chuỗi – thường các năm gần đây được dành làm tập kiểm tra, còn các năm cũ hơn dùng để huấn luyện, nhằm mô phỏng tình huống dự báo tương lai chưa từng thấy trong huấn luyện. Ngoài ra, chúng tôi đảm bảo rằng quá trình chuẩn hóa chỉ sử dụng thống kê của tập huấn luyện, và tập kiểm tra được chuẩn hóa dựa trên cùng tham số để đảm bảo tính khách quan.

Sau các bước trên, chúng tôi thu được một bộ dữ liệu huấn luyện hoàn chỉnh, sạch và được chuẩn hóa, sẵn sàng cho việc xây dựng và huấn luyện mô hình LSTM. Quá trình tiền xử lý này giúp đảm bảo rằng mô hình học được từ những tín hiệu thực trong dữ liệu (mối quan hệ giữa mưa và mực nước), thay vì học từ nhiễu hay bị ảnh hưởng bởi đơn vị đo lường khác nhau. Đây là bước nền tảng quan trọng để mô hình LSTM có thể hội tụ và dự báo chính xác.

2.4. Lựa chọn mô hình AI

2.4.1. Mạng Nơ-ron hồi tiếp LSTM (Long Short Term Memory)

a) Tổng quan về mạng LSTM và cấu trúc bộ nhớ dài – ngắn hạn

Long Short-Term Memory (LSTM) là một kiến trúc mạng nơ-ron hồi tiếp (RNN) đặc biệt, được Hochreiter & Schmidhuber đề xuất (1997) nhằm khắc phục nhược điểm "mất trí nhớ dài hạn" của RNN truyền thống. Khác với RNN thông thường chỉ có một trạng thái ẩn duy nhất truyền xuyên suốt thời gian, LSTM bổ sung thêm một bộ nhớ bên trong (cell state) cho phép lưu trữ thông tin trong thời gian dài, cùng với các cổng điều khiển

(gates) để quyết định luồng thông tin nào được lưu giữ hay quên đi qua các bước thời gian. Nhờ cấu trúc đặc biệt này, LSTM có khả năng học được các phụ thuộc dài hạn trong chuỗi mà không gặp phải hiện tượng độ dốc biến mất (vanishing gradient) trầm trọng như RNN truyền thống. Điều này làm cho LSTM đặc biệt hiệu quả trong các bài toán chuỗi thời gian phức tạp như dự báo lũ, nơi cần ghi nhớ ảnh hưởng tích lũy của mưa trong nhiều giờ/ngày.

Cấu trúc của một nút LSTM: Mỗi nút LSTM (tại một thời điểm t) gồm các thành phần chính: một ô nhớ (cell) lưu trữ trạng thái C_t , một trạng thái ẩn h_t (cũng chính là đầu ra của nút tại thời điểm t), và ba cổng điều khiển gọi là cổng đầu vào (i_t), cổng quên (f_t) và cổng đầu ra (o_t). Hình dung đơn giản, ô nhớ C_t giống như bộ nhớ chính lưu thông tin ngữ cảnh, còn trạng thái ẩn h_t là bộ nhớ ngắn hạn biểu thị đầu ra tức thời của ô. Ba cổng i, f, o (được implement dưới dạng các lớp neural với hàm kích hoạt sigmoid) đóng vai trò như những “van” điều tiết lượng thông tin chảy vào hoặc ra khỏi ô nhớ:

- Cổng quên (f_t): quyết định thông tin nào từ trạng thái cũ C_{t-1} cần bỏ đi. Cổng này nhận đầu vào là h_{t-1} (trạng thái ẩn tại bước trước) và x_t (đầu vào tại bước hiện tại), sau đó tính toán một hệ số f_t từ 0 đến 1 (qua hàm sigmoid) cho mỗi phần tử của C_{t-1} . Giá trị f_t gần 1 nghĩa là giữ lại thông tin cũ, gần 0 nghĩa là quên thông tin đó. Công thức toán học của cổng quên: $f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$ trong đó $[h_{t-1}, x_t]$ biểu diễn phép nối vector trạng thái ẩn cũ và đầu vào mới, W_f là ma trận trọng số của cổng quên và b_f là vector hệ số dịch. Kết quả f_t là một vector trong $(0,1)$ có cùng chiều với C_{t-1} , dùng để nhân từng phần tử với C_{t-1} sau đó (ký hiệu o_t).
- Cổng đầu vào (i_t): quyết định thông tin mới nào sẽ được lưu vào ô nhớ. Cổng này cũng nhận h_{t-1} và x_t , tính toán một vector i_t trong $(0,1)$ để điều tiết mức độ cho phép thông tin mới đi vào. Đồng thời, một lớp neural khác tạo ra vector ứng viên C_t chứa thông tin mới có thể thêm vào ô nhớ, thường sử dụng hàm kích hoạt tanh để giới hạn trong $[-1,1]$. Công thức: $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$, $C_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$, với W_i, W_c, b_i, b_c là trọng số và bias cho cổng đầu vào và cho bộ nhớ ứng viên tương ứng. Vector i_t quyết định từng phần thông tin mới

C_t sẽ được lưu bao nhiêu: giá trị i_t gần 1 cho phép thông tin tương ứng trong C_t được ghi vào ô nhớ nhiều hơn, và ngược lại.

- Cập nhật trạng thái ô nhớ (C_t): Sau khi có f_t và $i_t \square \tilde{C}_t$, trạng thái ô nhớ tại thời điểm t được cập nhật bằng cách kết hợp trạng thái cũ (đã quên một phần) và thông tin mới (đã chọn lọc): $C_t = f_t \square C_{t-1} + i_t \square \tilde{C}_t$, trong đó \square là phép nhân từng phần tử. Công thức này thể hiện rõ hai chức năng: (1) phần $f_t \square C_{t-1}$ là phần ký ức cũ được lưu giữ (nhờ công quên), (2) phần $i_t \square \tilde{C}_t$ là ký ức mới được thêm (nhờ công đầu vào). Nhờ cơ chế này, C_t có thể mang thông tin từ rất xa (nếu $f_t \square C_{t-1}$ qua nhiều bước) hoặc cập nhật nhanh với thông tin mới (nếu i_t cao và f_t giảm). Chính thiết kế này giúp giảm hiện tượng mất độ dốc: nếu cần nhớ lâu, mạng có thể học để $f_t \approx 1$, $i_t \approx 0$ qua nhiều bước, giữ C_t gần như không đổi.
- Công đầu ra (o_t): quyết định phần nào của trạng thái ô hiện tại C_t sẽ được đưa ra ngoài làm đầu ra h_t . Công đầu ra cũng nhận h_{t-1} và x_t , tính ra vector o_t trong $(0,1)$. Đồng thời, trạng thái ô C_t được đưa qua hàm tanh để chuyển thành một phiên bản “điều chỉnh” $\tanh(C_t)$ (giá trị trong $[-1,1]$). Cuối cùng, trạng thái ẩn mới (cũng chính là đầu ra của nút LSTM tại thời điểm t) được tính bằng: $h_t = o_t \square \tanh(C_t)$. Công thức này có nghĩa: công o_t sẽ đóng/mở các phần của $\tanh(C_t)$ để cho ra h_t . Nếu một phần tử của o_t gần 1, giá trị tương ứng từ ô nhớ (sau khi qua tanh) sẽ được xuất ra; nếu o_t gần 0, phần đó của C_t sẽ không ảnh hưởng đến h_t . Như vậy, o_t đóng vai trò bộ lọc đầu ra, đảm bảo chỉ những thông tin hữu ích nhất tại thời điểm hiện tại mới được truyền đến tầng tiếp theo hoặc bước thời gian tiếp theo dưới dạng h_t . Công thức tính công đầu ra: $o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$.

Tóm lại, tại mỗi bước thời gian t , một nút LSTM thực hiện các phép tính theo thứ tự: (1) tính f_t, i_t, C_t , (2) cập nhật C_t , và (3) tính o_t, h_t . Các công thức trên đây mô tả chi tiết luồng thông tin trong nút LSTM. Nhờ kiến trúc công như vậy, LSTM có khả năng lựa chọn học/nhớ hoặc quên thông tin một cách có kiểm soát. Ví dụ, nếu một thông tin mưa lớn cần được giữ để dự báo lũ vài ngày sau, mạng có thể học để công quên $f \approx 1$

giữ thông tin đó trong C_t qua nhiều bước; đồng thời nếu có nhiều nhiễu ngẫu nhiên, mạng có thể học để $\$i\$$ nhỏ, không lưu nhiều vào bộ nhớ.

2.4.2. Ưu điểm của LSTM so với RNN truyền thống

Mạng LSTM có một số ưu điểm vượt trội khi so sánh với mạng RNN truyền thống (vanilla RNN) trong bài toán dự báo lũ cũng như các bài toán chuỗi thời gian nói chung:

- Khả năng ghi nhớ dài hạn: Như đã trình bày, LSTM nhờ có cấu trúc ô nhớ và các cổng nên giảm thiểu được hiện tượng quên dần thông tin theo thời gian của RNN thường. RNN truyền thống gặp phải vấn đề vanishing gradient khiến mô hình hầu như không học được các phụ thuộc dài (ví dụ, mưa to 5 ngày trước ảnh hưởng đến lũ hôm nay) . LSTM cho phép dòng chảy gradient qua thời gian gần như không suy giảm nhờ cổng quên có thể giữ trạng thái gần như hằng số nếu cần . Do đó, LSTM học được các quan hệ dài hạn tốt hơn, rất phù hợp để dự báo lũ vì tác động tích lũy của mưa trong nhiều ngày mới gây lũ lớn.
- Giảm thiểu hiện tượng exploding gradient: Ngoài vanishing gradient, RNN còn có thể gặp exploding gradient (độ dốc bùng nổ) làm mất ổn định học. LSTM nhờ cơ chế cổng cũng phần nào giới hạn giá trị trạng thái trong phạm vi hàm $\$ \tanh \$$ và sigmoid ($[-1,1]$ và $[0,1]$), giúp giảm nguy cơ gradient quá lớn. Tuy LSTM không hoàn toàn loại bỏ được exploding gradient, nhưng trong thực tế huấn luyện LSTM thường ổn định hơn RNN thường.
- Chọn lọc thông tin quan trọng: Cấu trúc cổng cho phép LSTM lọc bỏ những thông tin nhiễu hoặc không quan trọng và chỉ giữ lại những yếu tố có ý nghĩa cho dự báo. Trong bài toán dự báo lũ, dữ liệu mưa và mực nước có thể chứa nhiều biến động ngẫu nhiên (ví dụ mưa nhỏ lát nhất không gây ảnh hưởng lớn). LSTM có thể học để bỏ qua (quên) những biến động nhỏ này, tập trung vào những trận mưa lớn có ý nghĩa. RNN thường không có khả năng chọn lọc như vậy, do đó có thể bị nhiễu bởi mọi biến động trong chuỗi.
- Hiệu quả đã được chứng minh trong các bài toán thời gian: LSTM là một trong những mô hình phổ biến nhất cho phân tích chuỗi thời gian, với nhiều ứng dụng thành công như nhận dạng giọng nói, dịch máy, phân tích chuỗi sự kiện và đặc biệt là dự báo chuỗi thời tiết thủy văn. Ưu thế của LSTM so với các phương pháp truyền thống (như mô hình vật lý hoặc các dạng hồi quy) trong dự báo lũ đã được chứng minh qua các nghiên cứu gần đây. Google Flood Hub – hệ thống dự báo lũ toàn cầu của Google – đã áp dụng mô hình LSTM để dự báo lưu lượng sông

lên đến 7 ngày với độ tin cậy cao. Mô hình LSTM của Google kết hợp dữ liệu mưa (mưa vệ tinh, dự báo thời tiết) và dữ liệu tĩnh lưu vực (độ dốc địa hình, loại đất, v.v.) để dự báo mức nước và đã đạt kết quả tương đương hoặc vượt trội so với các mô hình thủy văn truyền thống ở nhiều nơi trên thế giới. Nghiên cứu công bố trên tạp chí Nature (2024) cũng cho thấy mô hình AI dựa trên LSTM có thể dự báo các trận lũ cực hạn tại các lưu vực không có số liệu với độ tin cậy tương đương (thậm chí tốt hơn) so với hệ thống dự báo lũ vật lý hàng đầu (GloFAS). Điều này nghĩa là trí tuệ nhân tạo (LSTM) có thể cung cấp cảnh báo sớm hơn và bao phủ rộng hơn cho các sự kiện lũ lớn, kể cả tại những nơi thiếu trạm quan trắc. Nhiều nghiên cứu học thuật khác cũng áp dụng LSTM cho dự báo mưa lũ và thu được độ chính xác cao, khẳng định tính ưu việt của LSTM trong việc nắm bắt quan hệ phi tuyến giữa các biến khí tượng thủy văn.

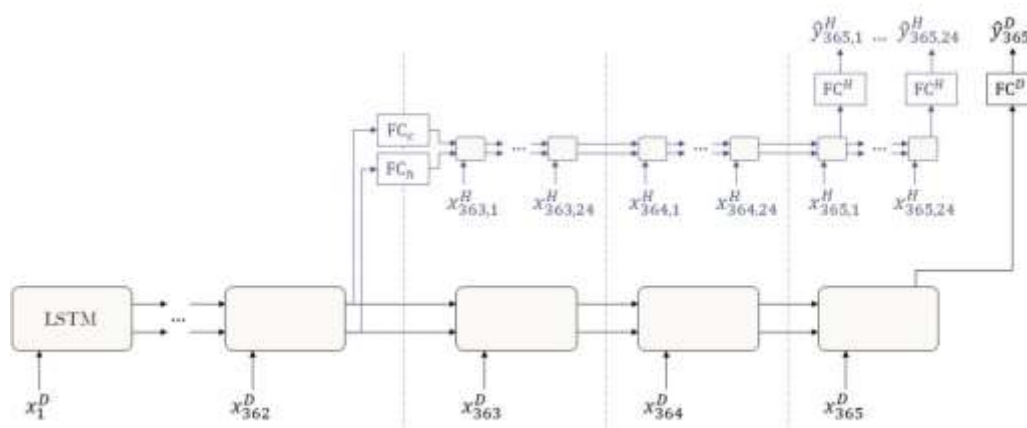
- Linh hoạt và mở rộng: LSTM có thể được mở rộng thành nhiều biến thể như Stacked LSTM (nhiều lớp LSTM chồng), Bi-directional LSTM (LSTM hai chiều), hoặc kết hợp với CNN thành CNN-LSTM, v.v. để tăng sức mạnh mô hình. Trong bài toán này, chúng tôi có thể xếp chồng 2–3 lớp LSTM để mô hình học được các đặc trưng phức tạp hơn từ dữ liệu mưa, mực nước. Nhờ tính linh hoạt, LSTM có thể tích hợp thêm nhiều đầu vào (ví dụ: lượng mưa dự báo, độ ẩm đất, v.v.) hoặc dự báo đa bước dễ dàng bằng cách điều chỉnh kiến trúc (ví dụ: dùng LSTM encoder-decoder để dự báo nhiều bước thời gian cùng lúc).
- Ưu điểm so với RNN về hiệu suất thực tế: Mặc dù LSTM phức tạp hơn (nhiều tham số hơn RNN thường), nhưng với sự hỗ trợ của phần cứng và thư viện học sâu tối ưu, việc huấn luyện LSTM hiện nay khá hiệu quả. LSTM tận dụng tốt song song hóa trên GPU do kiến trúc tính toán theo cổng. Ngoài ra, LSTM đã có sẵn trong nhiều thư viện (TensorFlow, PyTorch) với các tối ưu nội tại, do đó chi phí huấn luyện không còn là rào cản lớn so với trước đây.

Những ưu điểm nêu trên lý giải tại sao chúng tôi lựa chọn mô hình LSTM cho bài toán dự báo lũ. Trong bối cảnh dữ liệu mưa – lũ phức tạp tại Đà Nẵng, LSTM hứa hẹn nắm bắt được mối quan hệ thời gian dài hạn giữa mưa và mực nước, đồng thời đủ linh hoạt để bỏ qua các nhiễu ngắn hạn không quan trọng. Thực tế, các hệ thống dự báo hiện đại như Google Flood Hub đã minh chứng hiệu quả của LSTM trong dự báo lũ với độ chính xác và cảnh báo sớm vượt trội. Vì vậy, việc áp dụng LSTM trong luận văn này là hoàn toàn phù hợp và có cơ sở khoa học vững chắc.

2.5. Mô hình LSTM đa thang thời gian (MTS - LSTM)

Để cải thiện khả năng dự báo lũ (mức nước) ở cả tầm ngắn hạn (theo giờ) và trung hạn (theo ngày), đồ án sử dụng mô hình Long Short-Term Memory đa thang thời gian (Multi-Timescale LSTM - MTS-LSTM). MTS-LSTM kết hợp trong một kiến trúc hai nhánh LSTM: một nhánh xử lý dữ liệu thô ở thang ngày và một nhánh ở thang giờ. Cách tiếp cận này cho phép mô hình tận dụng được thông tin ở nhiều độ phân giải thời gian, vừa bao quát xu hướng dài ngày vừa chi tiết các biến động giờ, mà không cần chuỗi đầu vào quá dài như khi dùng LSTM đơn thang thời gian.

2.5.1. Kiến trúc tổng quan của mô hình MTS – LSTM



Hình 2.4. Kiến trúc mô hình LSTM đa thang thời gian (MTS - LSTM)

Sơ đồ kiến trúc MTS-LSTM với hai nhánh LSTM riêng cho hai thang thời gian. Nhánh ngày (màu đen) nhận chuỗi dữ liệu ngày và nhánh giờ (màu xanh) nhận chuỗi dữ liệu giờ. Tại điểm tách (ví dụ 14 ngày trước hiện tại trong hình), trạng thái cuối của nhánh ngày được chuyển qua nhánh giờ làm trạng thái khởi tạo (qua các lớp truyền trạng thái FC_c và FC_n). Mỗi nhánh có lớp đầu ra riêng (FC^D cho nhánh ngày và FC^H cho nhánh giờ) để dự báo mực nước ở từng thang thời gian tương ứng.

Mô tả kiến trúc: Mô hình gồm hai nhánh LSTM song song: nhánh LSTM ngày và nhánh LSTM giờ. Nhánh ngày xử lý chuỗi đầu vào ở độ phân giải ngày (ví dụ lượng mưa và mực nước tính theo ngày), còn nhánh giờ xử lý dữ liệu ở độ phân giải giờ. Giả sử ta muốn dự báo mực nước hiện tại (tại ngày d và giờ h trong ngày). Thay vì đưa toàn bộ dữ liệu quá khứ giờ (có thể lên đến hàng nghìn giờ) vào một LSTM đơn, MTS LSTM sẽ tách ra: nhánh ngày nhận thông tin dài hạn còn nhánh giờ nhận thông tin ngắn hạn:

- Nhánh LSTM ngày: đầu vào là chuỗi các giá trị theo ngày trong quá khứ xa (thường lấy tới 1 năm). Nhánh này tóm tắt xu hướng dài hạn của lưu vực (như mùa mưa, mùa khô, trạng thái tích lũy nước lưu vực). Sau khi xử lý T^D bước thời gian ngày, nhánh ngày tạo ra trạng thái ẩn cuối cùng h^D và trạng thái bộ nhớ c^D chứa thông tin tổng hợp của T^D ngày quá khứ.
- Nhánh LSTM giờ: đầu vào là chuỗi dữ liệu giờ gần thời điểm dự báo (ví dụ 3 ngày gần nhất tương ứng $T^H=72$ giờ). Nhánh này nắm bắt các biến động ngắn hạn, chẳng hạn các trận mưa trong vài ngày qua ảnh hưởng tức thời đến mực nước.

2.5.2. Cơ chế truyền trạng thái giữa các nhánh LSTM

Tại điểm giao giữa hai nhánh (ví dụ sau khi đã đọc 14 ngày dữ liệu quá khứ như minh họa ở Hình 2.1), mô hình thực hiện truyền trạng thái từ nhánh ngày sang nhánh giờ. Cụ thể, nếu h^D và c^D lần lượt là vector trạng thái ẩn và trạng thái bộ nhớ cuối cùng của nhánh LSTM ngày, ta sẽ chuyển đổi chúng thành trạng thái ban đầu cho nhánh $h_0^H = FC_h(h^D)$ và $c_0^H = FC_c(c^D)$ LSTM giờ: trong đó FC_h và FC_c là các lớp truyền trạng thái (thực chất là các lớp kết nối đầy đủ - fully connected). Hai lớp này có tác dụng ánh xạ vector trạng thái từ không gian nhánh ngày sang không gian nhánh giờ (nếu kích thước ẩn của hai LSTM khác nhau), đồng thời cho phép mô hình học cách biến đổi thông tin phù hợp với thang giờ.

Như vậy, cơ chế truyền trạng thái đảm bảo tính liên thông giữa hai nhánh: nhánh giờ không khởi động từ con số không mà được “khởi động có điều kiện” dựa trên thông tin dài hạn do nhánh ngày cung cấp. Điều này tương tự ý tưởng skip connection trong mạng ResNet, giúp giữ lại thông tin từ phần đầu mạng. Nhờ truyền trạng thái, mô hình MTS-LSTM có khả năng dự báo các đỉnh lũ ngắn hạn tốt hơn vì đã tích lũy được hiệu ứng của lượng mưa và dòng chảy từ nhiều ngày trước đó.

2.5.3. Đầu vào và đầu ra của từng nhánh LSTM

Mỗi nhánh LSTM trong mô hình MTS-LSTM có nhiệm vụ và định dạng đầu vào/ra khác nhau:

- Nhánh ngày (LSTM ngày):
 - + Đầu vào mỗi bước d là vector đặc trưng khí tượng thủy văn tính theo ngày (ví dụ: lượng mưa ngày d , mực nước trung bình ngày d , v.v.). Ký hiệu x_d^D là vector đầu vào ngày d .

+ LSTM ngày cập nhật trạng thái ẩn h_d^D và trạng thái bộ nhớ c_d^D qua công thức truy hồi LSTM tiêu chuẩn với đầu vào x_d^D . Sau T^D ngày, ta nhận được trạng thái cuối $H_{T^D}^D$ chứa thông tin tổng hợp.

+ Đầu ra dự báo ngày \hat{y}^D được tính thông qua một lớp kết nối đầy đủ như sau:

$$y^D = W_D h^D + b_D \quad (2.3)$$

Trong đó:

- W_D là ma trận trọng số
- b_D là hệ số điều chỉnh (bias) của lớp đầu ra cho nhánh ngày
- \hat{y}^D là giá trị mực nước dự báo ở thang thời gian ngày (ví dụ mực nước đỉnh ngày tiếp theo)

- Nhánh giờ (LSTM giờ)

+ Tại mỗi bước thời gian giờ t , đầu vào là vector đặc trưng X_t^H (ví dụ: lượng mưa tại giờ t , mực nước tại giờ t , v.v.).

+ LSTM giờ được khởi tạo với giá trị ban đầu:

$$(h_0^H, c_0^H) \quad (2.4)$$

+ Sau đó, nó nhận liên tiếp T^H bước giờ gần nhất và cập nhật trạng thái tại mỗi bước

$$(h_t^H, c_t^H) \quad (2.5)$$

+ Đầu ra dự báo giờ \hat{y}^H (ví dụ mực nước dự báo tại các giờ sắp tới) được tính từ trạng thái ẩn cuối $h_{T^H}^H$ qua một lớp kết nối đầy đủ riêng như sau:

$$\hat{y}^H = W_H h^H + b_H \quad (2.6)$$

Trong đó:

- W_H, b_H là tham số của lớp đầu ra cho nhánh giờ
- \hat{y}^H là giá trị mực nước dự báo theo giờ

+ Nếu dự báo nhiều bước thời gian giờ, lớp đầu ra có thể được thiết kế để xuất một vector (ví dụ 24 giá trị mực nước cho 24 giờ tương lai); trường hợp dự báo một bước thì \hat{y}^H là một giá trị tương ứng giờ kế tiếp.

Như vậy, hai nhánh LSTM hoạt động song song và cùng tạo dự báo ở hai độ phân giải. Mô hình có thể được huấn luyện với hàm mất mát tổng hợp từ sai số dự báo ngày và giờ. Đặc biệt, có thể bổ sung ràng buộc tính nhất quán liên thời gian: yêu cầu giá trị dự báo ngày phải xấp xỉ bằng trung bình các dự báo giờ trong ngày đó. Ràng buộc này

(thêm vào hàm loss dưới dạng regularization) giúp hai nhánh tạo kết quả nhất quán vật lý với nhau, tăng độ tin cậy của mô hình

2.5.4. Ưu điểm của mô hình MTS – LSTM so với LSTM thông thường

So với mô hình LSTM truyền thống (chỉ một nhánh thời gian), kiến trúc MTS-LSTM có nhiều ưu điểm trong bài toán dự báo lũ theo chuỗi thời gian:

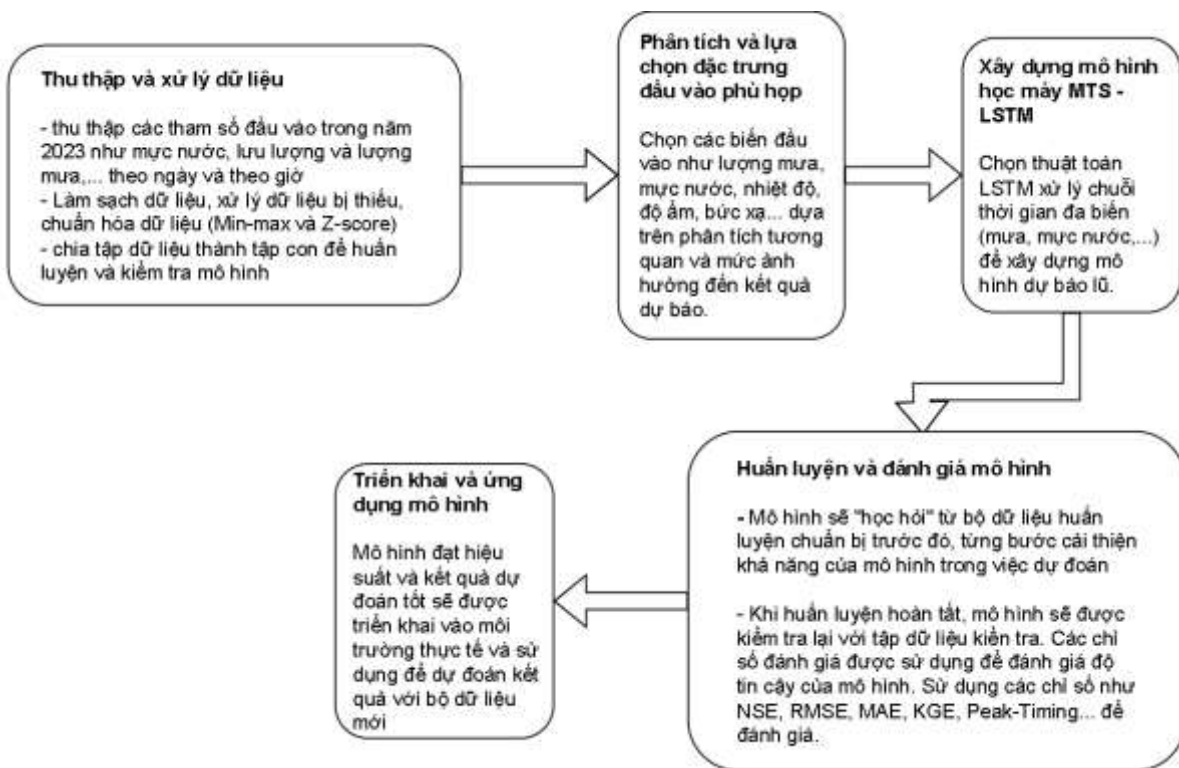
- **Học được cả xu hướng dài hạn và ngắn hạn:** Nhờ tách dữ liệu theo nhiều thang thời gian, MTS LSTM nắm bắt được các xu hướng dài ngày (mùa vụ, chu kỳ) đồng thời vẫn phản ánh được biến động ngắn hạn (như mưa giờ, lũ quét). LSTM thường nếu dùng dữ liệu giờ cho cả năm thì chuỗi đầu vào rất dài (~8760 giờ/năm), mô hình khó học quan hệ do quá nhiều bước thời gian và dẫn đến bất ổn định [33]. MTS-LSTM giải quyết bằng cách xử lý phần xa (quá khứ lâu) ở bước ngày cho gọn, sau đó chỉ dùng vài trăm bước giờ cho quá khứ gần [32, 33]. Nhờ đó giảm đáng kể độ dài chuỗi mỗi LSTM phải học, tăng tốc độ và tính ổn định khi huấn luyện.
- **Hiệu quả tính toán cao:** Thay vì huấn luyện hai mô hình riêng (một cho dự báo ngày, một cho giờ), MTS-LSTM gộp làm một mô hình duy nhất dự báo đồng thời hai mức thời gian. Cách này giúp tránh lặp thông tin và giảm chi phí tính toán, trong khi độ chính xác dự báo vẫn tương đương mô hình tách biệt [34]. Nghiên cứu của Gauch et al. (2021) cho thấy MTS-LSTM có hiệu năng ngang bằng việc dùng hai LSTM riêng lẻ, nhưng yêu cầu tài nguyên huấn luyện thấp hơn đáng kể [34].
- **Khai thác được dữ liệu đầu vào khác nhau cho mỗi thang:** Trong MTS-LSTM, mỗi nhánh có thể dùng bộ đặc trưng đầu vào khác nhau phù hợp với từng thang thời gian [34]. Ví dụ, nhánh ngày có thể dùng các biến khí tượng dự báo dài hạn (dự báo thời tiết 10 ngày, dữ liệu viễn thám cập nhật hàng ngày), còn nhánh giờ dùng dữ liệu mưa đo thực tế hàng giờ. LSTM thông thường không thể đồng thời sử dụng hai nguồn dữ liệu có tần suất khác nhau một cách hiệu quả, trong khi MTS-LSTM được thiết kế cho mục đích này [34]. Điều này rất hữu ích trong bối cảnh dự báo lũ, khi mà dự báo mưa dài hạn thường có độ phân giải ngày, còn mưa ngắn hạn thì có dữ liệu chi tiết theo giờ

Tóm lại, kiến trúc MTS-LSTM được lựa chọn vì phù hợp với bài toán dự báo mực nước đa thang thời gian: nó vừa tận dụng được dữ liệu dài hạn vừa tập trung vào tín hiệu ngắn hạn quan trọng cho lũ, giúp cải thiện độ chính xác dự báo và hiệu quả mô

hình so với LSTM thông thường. Các ưu điểm về tính linh hoạt và nhất quán thời gian của MTS-LSTM khiến mô hình này trở thành một giải pháp tiềm năng cho hệ thống cảnh báo lũ hiện đại.

Tài liệu tham khảo: Mô hình MTS-LSTM được tham khảo và điều chỉnh từ nghiên cứu của Gauch et al. (2021) [33, 34] và tài liệu hướng dẫn của nhóm Neural Hydrology [33], trong đó đã chứng minh hiệu quả của kiến trúc này trên dữ liệu dự báo dòng chảy nhiều lưu vực. Các kỹ thuật tiền xử lý dữ liệu được áp dụng theo thông lệ trong lĩnh vực khai thác dữ liệu thời gian thực và được điều chỉnh cho phù hợp với bộ dữ liệu trạm Đà Nẵng.

2.6. Mô hình truyền thống Cloud AI



Hình 2.5. Sơ đồ mô hình kiến trúc truyền thống Cloud AI

Chương 3: EDGE AI – HƯỚNG TIẾP CẬN MỚI TRONG DỰ BÁO VÀ CẢNH BÁO LŨ LỤT

3.1. Lí do ứng dụng Edge AI trong bối cảnh lũ lụt đô thị

Trong những năm gần đây, Đà Nẵng đối mặt với tình trạng ngập lụt ngày càng nghiêm trọng do tác động của biến đổi khí hậu, đô thị hóa nhanh và hệ thống thoát nước xuống cấp. Ngoài ra, việc xả lũ từ các hồ thủy điện thượng nguồn càng làm gia tăng nguy cơ lũ lụt, đòi hỏi thành phố cần có hệ thống dự báo và cảnh báo hiệu quả hơn.

Hiện nay, thành phố có mạng lưới trạm quan trắc mực nước và lượng mưa thu thập dữ liệu liên tục. Tuy nhiên, việc truyền tải toàn bộ dữ liệu thô về trung tâm không chỉ gây quá tải băng thông mà còn làm giảm hiệu suất dự báo. Do đó, cần một giải pháp tối ưu hơn trong xử lý dữ liệu.

Edge AI – công nghệ trí tuệ nhân tạo tích hợp tại thiết bị biên – mang đến hướng đi mới. Thay vì phụ thuộc hoàn toàn vào máy chủ trung tâm, Edge AI cho phép xử lý dữ liệu ngay tại hiện trường, phát hiện sớm các bất thường như mưa lớn, mực nước dâng và đưa ra cảnh báo tức thời. Công nghệ này đặc biệt hữu ích ở các khu vực dễ bị cô lập khi thiên tai xảy ra, đảm bảo hệ thống cảnh báo hoạt động liên tục và tin cậy.

Ngoài khả năng phản ứng nhanh, Edge AI còn hỗ trợ xây dựng các mô hình dự báo riêng cho từng khu vực như Hòa Khánh hay Cẩm Lệ, dựa trên dữ liệu và đặc điểm thủy văn địa phương, giúp tăng độ chính xác và phù hợp với thực tế. Đồng thời, việc xử lý dữ liệu tại chỗ giúp giảm áp lực lên trung tâm dữ liệu, tiết kiệm chi phí truyền thông, bảo vệ thông tin nhạy cảm, đặc biệt khi thu thập hình ảnh hoặc video trong khu dân cư.

Ứng dụng Edge AI trong quản lý ngập lụt thể hiện sự chuyển đổi từ phản ứng bị động sang phòng ngừa chủ động, từ tập trung sang phân tán, góp phần xây dựng một hệ sinh thái đô thị thông minh và bền vững. Tuy nhiên, để triển khai hiệu quả cần đầu tư đồng bộ vào hạ tầng, kỹ thuật và nhân lực, cũng như sự phối hợp giữa chính quyền, doanh nghiệp, viện nghiên cứu và cộng đồng.

3.2. So sánh Edge AI với mô hình tập trung truyền thống

3.2.1. Ưu điểm của Edge AI

Giảm độ trễ xử lý và phản hồi nhanh

- Trong mô hình tập trung, dữ liệu được truyền từ cảm biến lên máy chủ trung tâm để xử lý, gây ra độ trễ lớn, đặc biệt khi mạng bị nghẽn hoặc mất kết nối.

- Edge AI xử lý dữ liệu ngay tại điểm thu thập (thiết bị biên), giúp ra quyết định gần như tức thì, đảm bảo cảnh báo kịp thời để người dân và chính quyền ứng phó hiệu quả.

Tiết kiệm băng thông và chi phí truyền tải dữ liệu

- Truyền toàn bộ dữ liệu thô về trung tâm tiêu tốn băng thông lớn, tăng chi phí vận hành mạng và có thể làm nghẽn hệ thống khi lượng dữ liệu lớn.
- Edge AI chỉ truyền dữ liệu đã được xử lý hoặc các cảnh báo quan trọng, giảm tải băng thông, tiết kiệm chi phí vận hành hệ thống mạng và đám mây.

Tăng cường bảo mật và bảo vệ dữ liệu cá nhân

- Dữ liệu nhạy cảm (như hình ảnh, video, dữ liệu vị trí) khi xử lý tập trung dễ bị rò rỉ hoặc tấn công mạng trong quá trình truyền.
- Xử lý tại biên giúp hạn chế truyền tải dữ liệu nhạy cảm, nâng cao tính bảo mật và giảm nguy cơ bị khai thác thông tin trái phép.

Tính sẵn sàng cao và khả năng vận hành độc lập

- Hệ thống tập trung phụ thuộc hoàn toàn vào kết nối mạng và trung tâm xử lý; nếu mất kết nối, toàn bộ hệ thống có thể bị gián đoạn.
- Các thiết bị Edge AI có thể hoạt động độc lập, tiếp tục xử lý và đưa ra cảnh báo dù mạng bị gián đoạn, giúp hệ thống bền bỉ hơn trong điều kiện thiên tai.

Khả năng mở rộng và linh hoạt cao

- Mô hình tập trung gặp khó khăn khi mở rộng quy mô, do tăng tải lên trung tâm xử lý và mạng truyền dữ liệu.
- Edge AI cho phép triển khai linh hoạt tại nhiều điểm biên khác nhau, dễ dàng mở rộng hệ thống theo vùng địa lý, đáp ứng đa dạng điều kiện địa phương.

Tối ưu hóa tài nguyên tính toán và năng lượng

- Xử lý dữ liệu phân tán giúp tận dụng hiệu quả các thiết bị vi xử lý biên nhỏ gọn, tiết kiệm năng lượng so với việc gửi toàn bộ dữ liệu về trung tâm.
- Giúp giảm tải cho các trung tâm dữ liệu lớn, kéo dài tuổi thọ thiết bị và giảm chi phí vận hành.

3.2.2. So sánh định lượng qua các chỉ số

Việc lựa chọn kiến trúc Edge AI trong hệ thống dự báo lũ lụt không chỉ xuất phát từ yêu cầu thực tế về môi trường triển khai, mà còn được củng cố thông qua các phân tích định lượng và lập luận khoa học rõ ràng.

a) Độ trễ xử lý (Latency)

Trước tiên, về **độ trễ (latency)**, xử lý tại Cloud yêu cầu tổng thời gian gồm ba thành phần: thời gian truyền dữ liệu từ hiện trường lên Cloud (T_{upload}), thời gian suy luận mô hình ($T_{inference}$) và thời gian phản hồi lại thiết bị ($T_{download}$), tạo thành công thức tổng [49]:

Cloud AI:

$$T_{Cloud} = T_{upload} + T_{inference} + T_{download} \quad (3.1)$$

Trong đó:

- T_{upload} : thời gian truyền dữ liệu từ trạm đến server
- $T_{inference}$: thời gian server xử lý mô hình AI
- $T_{download}$: thời gian phản hồi kết quả về trạm

Trong khi đó, với kiến trúc Edge AI, toàn bộ quá trình xử lý diễn ra tại chỗ, rút ngắn thời gian xuống chỉ còn:

Edge AI:

$$T_{edge} = T_{local-inference} \quad (3.2)$$

So sánh: Trên thực tế, T_{Cloud} thường rơi vào khoảng 1–3 giây, trong khi Edge AI có thể xử lý trong dưới 100 mili-giây.

⇒ Edge AI đáp ứng tốt hơn các yêu cầu thời gian thực trong cảnh báo thiên tai.

Với kết quả như vậy hệ thống có thể đưa ra cảnh báo nhanh hơn, rất cần thiết trong các tình huống khẩn cấp như mực nước dâng đột ngột.

b) Tính lượng băng thông cần thiết để truyền toàn bộ dữ liệu có biến động từ 2 tập dữ liệu:

Trong hệ thống dự báo lũ lụt sử dụng AI, lượng dữ liệu cảm biến cần được truyền thường xuyên từ hiện trường về trung tâm xử lý là rất lớn, đặc biệt với các cảm biến mực nước và lượng mưa đo theo chu kỳ 1 giờ. Theo công thức cơ bản [50]:

Cloud AI:

$$B_{cloud} = n.s \quad (3.3)$$

trong đó n là số bản ghi dữ liệu trong ngày và s là kích thước trung bình mỗi bản ghi, có thể ước tính tổng băng thông cần thiết để gửi toàn bộ dữ liệu thô (raw) lên Cloud.

Ví dụ: với hai tập dữ liệu thực tế tại hồ Thạch Gián năm 2023:

- Dữ liệu lượng mưa: 8,752 bản ghi, ~0.14 MB → trung bình ~16 byte/bản ghi
- Dữ liệu mực nước: 11,735 bản ghi, ~0.23 MB → trung bình ~20 byte/bản ghi

⇒ Tổng dữ liệu mỗi năm: ~0.37 MB

Giả sử hệ thống truyền dữ liệu thô lên Cloud toàn bộ (không lọc), băng thông mỗi lần đồng bộ sẽ ở mức **~0.37 MB/lần**.

Tuy nhiên, nếu sử dụng Edge AI để lọc và xử lý tại chỗ, chỉ truyền khi có "biến động lớn" hoặc vượt ngưỡng, ta có thể tiết kiệm băng thông đáng kể. Phân tích dữ liệu cho thấy:

- Chỉ **6.56%** dữ liệu lượng mưa và **0.55%** dữ liệu mực nước có biến động lớn
- Khi đó, dung lượng cần gửi chỉ còn:
 - + Lượng mưa: $0.0656 \times 0.14 \approx 0.0092$ MB
 - + Mực nước: $0.0055 \times 0.23 \approx 0.0013$ MB
 - + **Tổng cộng: ~0.0105 MB**

Như vậy, hệ thống Edge AI giúp **giảm băng thông truyền tải hơn 97%**, đồng thời giảm tải xử lý ở máy chủ trung tâm và tăng tính phản hồi trong thời gian thực.

Vậy, Nếu lọc dữ liệu ngay tại trạm và chỉ truyền các bản ghi có dấu hiệu biến động lớn về lượng mưa hoặc mực nước, ta có thể giảm dung lượng dữ liệu cần truyền từ 0.37 MB xuống chỉ còn khoảng 0.01 MB, tức là tiết kiệm hơn 97 % băng thông và giảm đáng kể thời gian xử lý server

c) **Năng lượng tiêu thụ**

Về **năng lượng tiêu thụ**, kiến trúc Edge cũng tỏ ra hiệu quả hơn trong môi trường không có nguồn điện ổn định. Dựa vào công thức vật lý:

$$E = \left(\frac{V^2}{R} \right) . t \quad (3.4)$$

Trong đó

- E : là năng lượng tiêu thụ khi truyền dữ liệu
- V : là điện áp cung cấp
- R : là điện trở của thiết bị đang tiêu thụ điện
- t : thời gian truyền nhận dữ liệu

⇒ Cloud AI truyền nhận liên tục nên t lớn ⇒ tiêu hao nhiều năng lượng.

⇒ Edge AI xử lý cục bộ, không cần truyền thường xuyên ⇒ **tiết kiệm năng lượng**, phù hợp với nguồn cấp từ pin mặt trời hoặc acquy.

Qua đó, việc truyền dữ liệu liên tục và thực thi AI từ xa sẽ tiêu tốn nhiều năng lượng hơn so với xử lý tại chỗ bằng các vi điều khiển tiết kiệm điện như STM32 hay ESP32.

d) Chi phí vận hành

Về **chi phí vận hành tổng thể**, theo mô hình TCO (Total Cost of Ownership), tổng chi phí bao gồm chi phí phần cứng, chi phí truyền thông (data/network), và chi phí bảo trì hệ thống. Dù thiết bị Edge AI có thể cao hơn về chi phí ban đầu, nhưng nhờ giảm được lưu lượng dữ liệu, đơn giản hóa hạ tầng mạng và bảo trì, tổng chi phí vận hành sẽ thấp hơn đáng kể về lâu dài [51]:

Cloud AI:

$$C_{cloud} = C_{server} + C_{compute} + C_{storage} + C_{network} \quad (3.5)$$

Trong đó:

- C_{cloud} : Tổng chi phí vận hành của Cloud AI
- C_{server} : Chi phí phần cứng / máy chủ của Cloud
- $C_{compute}$: Chi phí tính toán (computational cost) của Cloud
- $C_{storage}$: Chi phí lưu trữ dữ liệu trong Cloud
- $C_{network}$: Chi phí truyền tải dữ liệu mạng trong Cloud

Edge AI:

$$C_{edge} = C_{element} + C_{maintain} \quad (3.6)$$

Trong đó

- C_{edge} : Tổng chi phí vận hành của Edge AI
- $C_{element}$: Chi phí phần tử triển khai (thiết bị tại điểm nút – ví dụ cảm biến thông minh, bo mạch...)

- $C_{maintain}$: Chi phí bảo trì thiết bị Edge sau triển khai

e) **Bảo mật và quyền riêng tư**

Yếu tố **bảo mật và độ tin cậy**, vốn rất quan trọng trong các hệ thống cảm biến môi trường ngoài trời. Khi xử lý tại Cloud, toàn bộ dữ liệu phải rời khỏi thiết bị, làm tăng nguy cơ rò rỉ thông tin và phụ thuộc hoàn toàn vào kết nối mạng. Trong khi đó, với kiến trúc Edge, dữ liệu được xử lý nội bộ và chỉ truyền phần cần thiết, giúp giảm thiểu rủi ro bảo mật và tăng tính ổn định hệ thống trong điều kiện thời tiết xấu hoặc sự cố đường truyền [52].

3.2.3. **Kết luận**

Với dữ liệu lượng mưa và mực nước hồ Thạch Gián được thu thập liên tục theo thời gian, việc triển khai mô hình dự báo AI trực tiếp tại thiết bị biên (Edge) giúp:

- Rút ngắn độ trễ
- Giảm thiểu băng thông
- Tiết kiệm năng lượng
- Tối ưu chi phí dài hạn
- Nâng cao bảo mật

Do đó, **Edge AI là lựa chọn vượt trội hơn so với Cloud AI** cho bài toán cảnh báo ngập đô thị theo thời gian thực tại Đà Nẵng là **có khả thi**.

3.3. **Lựa chọn thiết bị Edge AI phù hợp**

Trong hệ thống dự báo và cảnh báo lũ lụt tại khu vực hồ Thạch Gián, việc tích hợp một thiết bị Edge AI tại hiện trường là cần thiết nhằm đảm bảo khả năng xử lý dữ liệu ban đầu một cách nhanh chóng, giảm độ trễ truyền tải, cũng như tiết kiệm băng thông và chi phí vận hành. Qua phân tích yêu cầu thực tế của hệ thống, dữ liệu cảm biến thu thập liên tục bao gồm mực nước và lượng mưa theo chu kỳ 5 phút, dẫn đến việc hình thành một tập dữ liệu có kích thước lớn theo thời gian. Việc xử lý toàn bộ dữ liệu này tại nền tảng Cloud là không tối ưu, đặc biệt trong bối cảnh mất kết nối mạng thường xuyên xảy ra trong mùa mưa bão.

Để giải quyết bài toán này, thiết bị Edge AI sẽ được triển khai nhằm thực hiện các tác vụ tại chỗ như thu thập và tiền xử lý dữ liệu, loại bỏ nhiễu, tính toán các chỉ số đơn giản như lượng mưa tích lũy và phát hiện các tình huống vượt ngưỡng nguy hiểm (ví dụ như mực nước tăng nhanh đột ngột). Trong trường hợp các điều kiện bất thường

được phát hiện, thiết bị sẽ chủ động gửi cảnh báo hoặc chỉ truyền những gói dữ liệu cần thiết lên Cloud để thực hiện phân tích sâu hơn và dự báo bằng mô hình AI.

Do tính chất phức tạp và yêu cầu tài nguyên cao của các mô hình học sâu như LSTM – vốn được sử dụng để dự báo lũ dựa trên chuỗi thời gian dài – nhóm đánh giá rằng các vi điều khiển nhẹ như ESP32 hay STM32 không đáp ứng đủ khả năng tính toán để suy luận các mô hình này tại chỗ. Vì vậy, các thiết bị Edge trong hệ thống **chỉ đảm nhận vai trò “lọc trước”**, giúp **giảm đáng kể lượng dữ liệu truyền đi và giảm tải cho hệ thống Cloud**. Mô hình AI phức tạp sẽ được thực thi hoàn toàn tại phía Cloud, nơi có đủ tài nguyên để huấn luyện, cập nhật, và suy luận mô hình một cách hiệu quả.

Trên cơ sở đó, nhóm đề xuất sử dụng thiết bị Edge AI ở cấp vi điều khiển có hiệu năng tương đối cao như **Raspberry Pi 4** để triển khai tại các trạm quan trắc. Raspberry Pi 4 đáp ứng tốt nhu cầu xử lý dữ liệu cảm biến, phân tích tại chỗ và truyền dữ liệu cần thiết về trung tâm. Việc lựa chọn một nền tảng phần cứng thống nhất giúp hệ thống vận hành ổn định, dễ mở rộng và tối ưu hóa chi phí đầu tư cho hạ tầng truyền thông và tính toán.



Hình 3.1. Lựa chọn thiết bị Edge

3.4. Thiết kế tổng quan hệ thống Edge AI trong dự báo lũ lụt

Hệ thống cảnh báo ngập lụt ứng dụng Edge AI được thiết kế với cấu trúc phân tầng, trong đó nhiệm vụ thu thập và xử lý ban đầu được thực hiện tại thiết bị biên (Edge), còn việc dự đoán và phát cảnh báo được xử lý trên nền tảng điện toán đám mây (Cloud). Mô hình này cho phép tận dụng hiệu quả cả hai môi trường Edge và Cloud, đảm bảo tính linh hoạt, giảm độ trễ xử lý và tối ưu tài nguyên hệ thống.

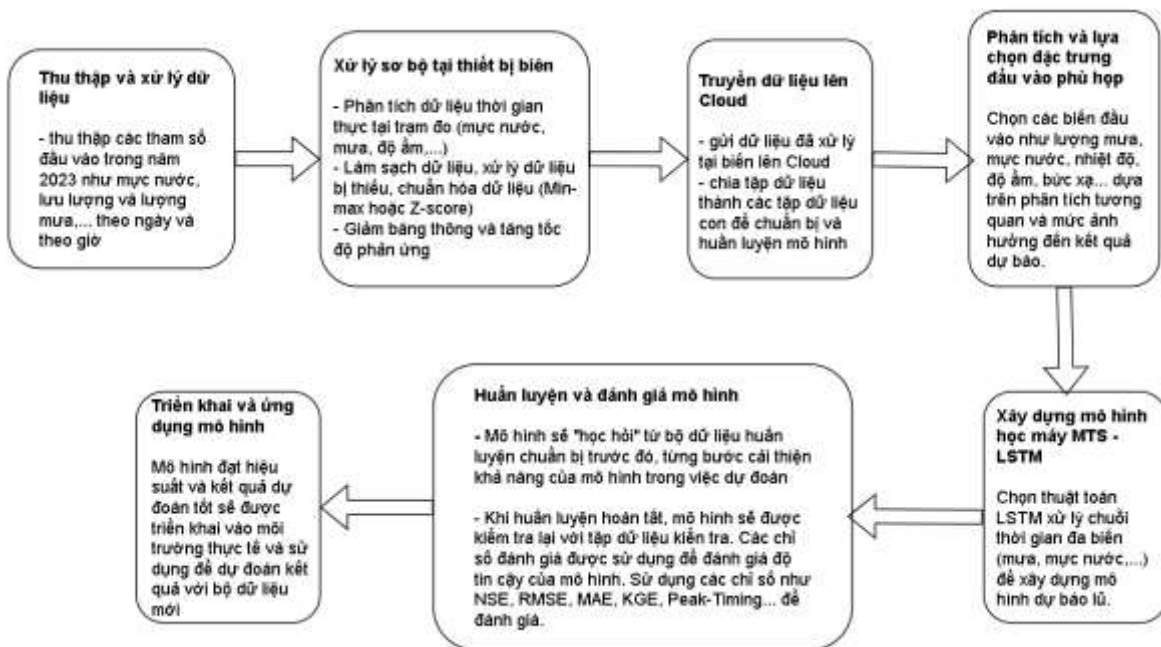
3.4.1. Kiến trúc tổng thể hệ thống

Hệ thống được chia thành hai thành phần chính:

- **Thiết bị Edge:** Gồm vi điều khiển hoặc máy tính nhúng có khả năng thu thập dữ liệu từ cảm biến (mức nước, lượng mưa,...), lưu trữ dữ liệu dưới dạng tệp CSV và thực hiện tiền xử lý dữ liệu (lọc nhiễu, xử lý thiếu, chuẩn hóa,...). Sau khi xử lý, dữ liệu được gửi lên Cloud thông qua mạng WiFi hoặc 4G.
- **Hệ thống Cloud AI:** Nhận dữ liệu đã được xử lý từ Edge, thực hiện dự đoán mực nước bằng mô hình học sâu (Deep Learning). Sau đó, kết quả được so sánh với các ngưỡng cảnh báo được thiết lập sẵn. Nếu vượt ngưỡng, hệ thống sẽ lưu log và gửi cảnh báo qua email hoặc SMS đến người dùng.

3.4.2. Sơ đồ mô hình kiến trúc phân tán Edge – Cloud

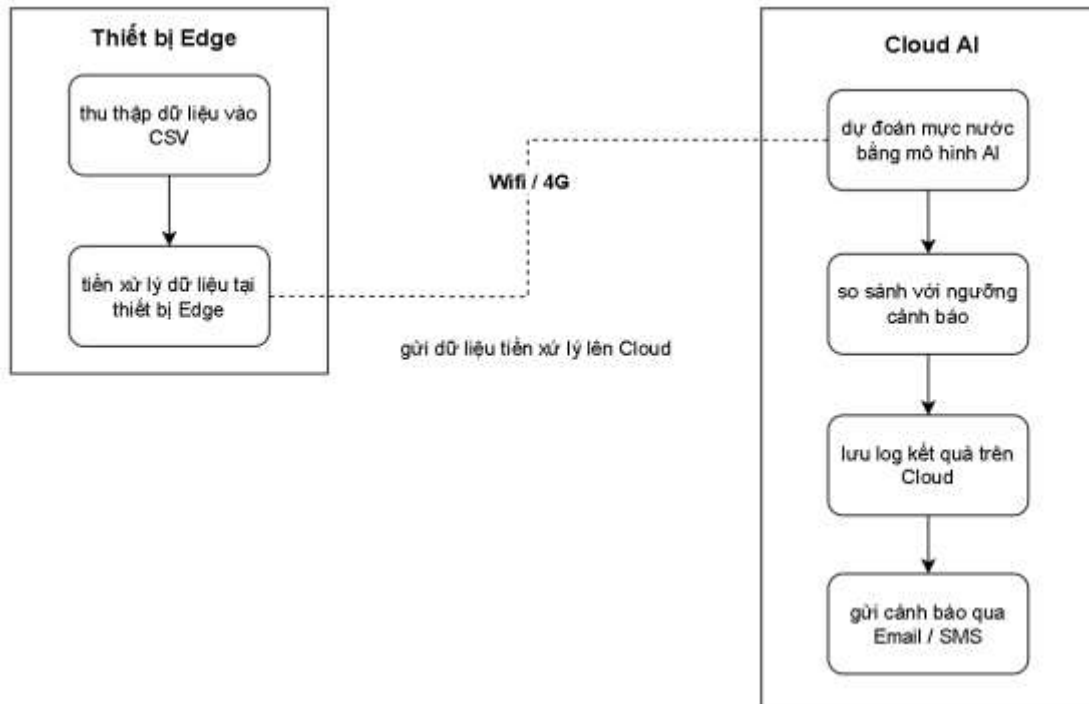
Hình 3.2 dưới đây minh họa kiến trúc tổng quan của hệ thống, thể hiện rõ chức năng của từng thành phần và luồng dữ liệu từ cảm biến đến đầu ra cảnh báo:



Hình 3.2. Sơ đồ mô hình kiến trúc phân tán Edge – Cloud

3.4.3. Sơ đồ luồng dữ liệu trong mô hình mô phỏng

Trong giai đoạn triển khai mô phỏng, dữ liệu cảm biến được thay thế bằng file CSV và xử lý trực tiếp trên máy tính thay vì thiết bị phần cứng. Sơ đồ 3.3 trình bày cụ thể chu trình luồng dữ liệu từ lúc thu thập, xử lý, dự đoán đến khi gửi cảnh báo:



Hình 3.3. Sơ đồ luồng dữ liệu của hệ thống Edge AI trong mô hình mô phỏng

3.5. Triển khai hệ thống Edge AI

Trong bối cảnh giới hạn về thời gian, nguồn lực và điều kiện thực tế triển khai, nhóm nghiên cứu quyết định không thực hiện phần cứng vật lý cho hệ thống dự báo - cảnh báo ngập lụt, mà thay vào đó mô phỏng toàn bộ quy trình xử lý dữ liệu và ứng dụng trí tuệ nhân tạo trên môi trường phần mềm. Cách tiếp cận này vẫn đảm bảo thể hiện rõ nguyên lý hoạt động của một hệ thống Edge AI và chứng minh tính khả thi của mô hình khi được triển khai thực tế trong tương lai.

3.5.1. Lý do không triển khai phần cứng

- **Giới hạn về thời gian và thiết bị:** Việc triển khai phần cứng yêu cầu các thiết bị như cảm biến mưa, cảm biến mực nước, máy tính nhúng (Raspberry Pi 4), mô-đun giao tiếp (UART, WiFi), nguồn điện ổn định,... Những thành phần này cần được mua sắm, lập trình, kiểm thử và hiệu chuẩn, đòi hỏi nhiều thời gian và công sức.
- **Tập trung vào cốt lõi thuật toán AI:** Mục tiêu chính của đề tài là nghiên cứu và xây dựng mô hình AI có khả năng dự đoán ngập lụt chính xác, từ đó kết hợp với giao diện cảnh báo. Việc mô phỏng phần cứng bằng dữ liệu thật và mô phỏng đường truyền giúp tối ưu hóa thời gian để hoàn thiện lõi xử lý thông minh.

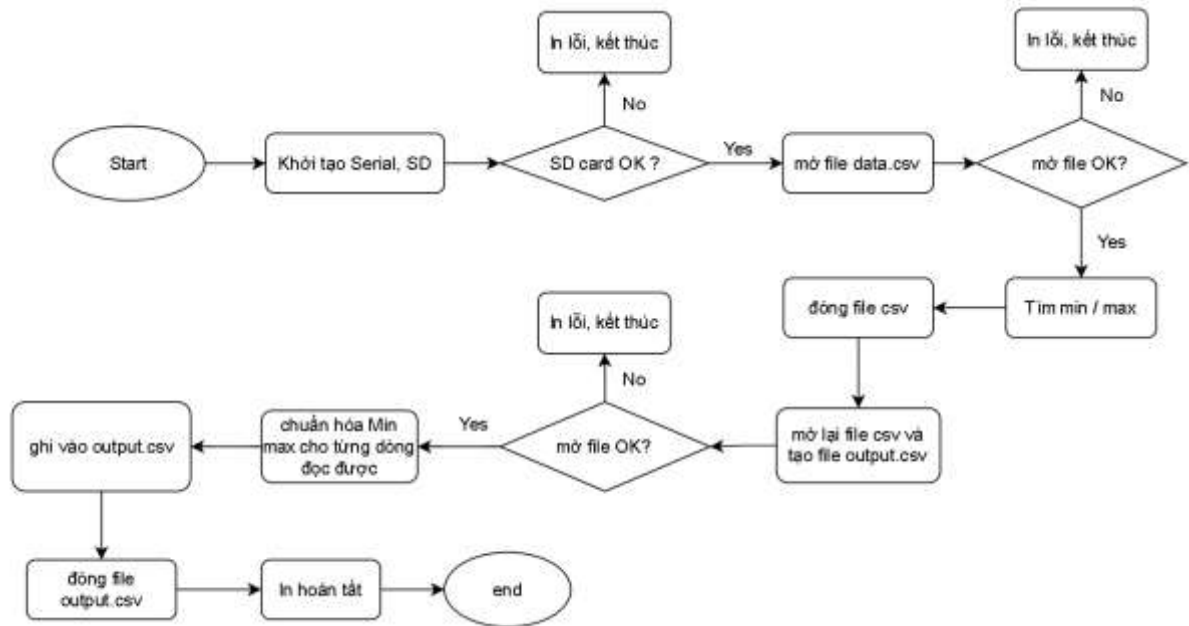
- **Mở rộng dễ dàng về sau:** Toàn bộ hệ thống được thiết kế mở, các thành phần đều có thể tích hợp với phần cứng thực trong tương lai mà không cần viết lại từ đầu.

3.5.2. Kiến trúc mô phỏng hệ thống Edge AI

Trong trường hợp không có phần cứng thực tế, nhóm đã xây dựng một mô hình mô phỏng hệ thống Edge AI theo kiến trúc như sau:

- **Dữ liệu mô phỏng cảm biến:** Các file .csv chứa dữ liệu mưa, mực nước theo thời gian thực tế tại hồ Thạc Gián, Đà Nẵng được sử dụng như đầu vào thay cho cảm biến vật lý. Dữ liệu này được xem như dữ liệu đầu ra của các cảm biến thực tế.
- **Tiền xử lý dữ liệu tại thiết bị Edge (Vi điều khiển):** Vi điều khiển (ESP32) được lập trình để đọc dữ liệu từ file data.csv trên thẻ SD, thực hiện
 - + Tính giá trị min và max cho từng đặc trưng (mực nước và lượng mưa)
 - + Chuẩn hóa Min – Max từng dòng dữ liệu
 - + Ghi dữ liệu đã chuẩn hóa vào file mới output.csv trên thẻ SD
 - + (Tùy chọn): gửi từng dòng dữ liệu đã chuẩn hóa qua UART tương ứng cho quá trình gửi dữ liệu từ thiết bị biên lên cloud (chưa triển khai trong mô phỏng)
- **Dự báo bằng mô hình AI:** File output.csv sẽ được đưa lên môi trường Cloud (Google Colab), nơi mô hình học sâu (Multi - time-step LSTM) được triển khai để dự báo mực nước trong tương lai.
- **Hệ thống cảnh báo:** Nếu kết quả dự báo vượt ngưỡng, hệ thống sẽ kích hoạt cảnh báo qua console hoặc gửi email tự động. Điều này mô phỏng hoạt động của một thiết bị cảnh báo như còi, đèn báo ngoài thực tế. (chưa triển khai trong mô phỏng)

Luồng xử lý này được thể hiện trong **Hình 3.3 – Sơ đồ luồng dữ liệu của hệ thống Edge AI trong mô hình mô phỏng (3.4.3)** và thuật toán xử lý dữ liệu trực tiếp trên vi điều khiển được thể hiện trong **Hình 3.4 – Lưu đồ thuật toán xử lý dữ liệu tại thiết bị Edge** như hình dưới:



Hình 3.4. Lưu đồ thuật toán tiền xử lý dữ liệu tại thiết bị biên

Sau khi hoàn tất quá trình tiền xử lý (loại nhiễu, chuẩn hóa Min - Max), vi điều khiển ghi dữ liệu đã xử lý vào file output.csv trên thẻ SD.

File này mô phỏng vai trò truyền dữ liệu lên Cloud, và được sử dụng làm đầu vào cho mô hình học sâu

3.5.3. So sánh giữa mô phỏng và phần cứng

Bảng 3.1. So sánh kiểm nghiệm trên mô phỏng và phần cứng

Thành phần	Khi có phần cứng	Mô phỏng trong báo cáo
Cảm biến vật lý	Dữ liệu từ thiết bị thực (analog / digital)	File .csv dữ liệu thời gian thực
Vi điều khiển	STM32 / Raspberry Pi 4 thu thập và xử lý sơ bộ	Google Colab xử lý toàn bộ
Giao tiếp dữ liệu	UART, WiFi hoặc LoRa truyền về Cloud	Dữ liệu nạp trực tiếp vào notebook
Mô hình AI	Nhúng hoặc gọi từ cloud	Chạy trực tiếp trên Colab
Hành động cảnh báo	Điều khiển thiết bị thực (đèn, còi, relay)	Hiển thị console hoặc gửi email

Như vậy, dù không có phần cứng thực tế, hệ thống vẫn thể hiện đầy đủ luồng xử lý thông tin và cảnh báo, bảo đảm logic Edge AI vẫn được duy trì.

3.5.4. Tính mở rộng về sau:

Mặc dù hệ thống hiện tại chưa có phần cứng, kiến trúc được thiết kế mở và có thể dễ dàng tích hợp thiết bị thực tế trong tương lai:

- **ESP32** có thể xử lý tín hiệu cảm biến và gửi dữ liệu lên cloud thông qua WiFi.
- **STM32** có thể đọc tín hiệu analog từ cảm biến mực nước và gửi đi qua UART hoặc LoRa.
- **Mô hình AI** có thể được tối ưu hóa với TensorFlow Lite để chạy trực tiếp trên vi điều khiển nếu đủ tài nguyên.
- **Kịch bản đa điểm:** Có thể mở rộng theo mô hình nhiều trạm đo gửi dữ liệu về trung tâm xử lý hoặc thực hiện cảnh báo tại chỗ.

3.5.5. Kết luận

Trong Chương 3, việc mô phỏng hệ thống Edge AI thay vì triển khai phần cứng thực tế đã giúp nhóm tập trung tối ưu vào việc phát triển, kiểm thử và đánh giá hiệu quả của mô hình AI một cách chính xác và linh hoạt. Bên cạnh đó, nhóm đã thiết kế kiến trúc hệ thống mở, có khả năng tích hợp dễ dàng với phần cứng thật trong tương lai khi điều kiện cho phép. Điều này không chỉ khẳng định tính khả thi của giải pháp về mặt thuật toán, mà còn cho thấy tiềm năng ứng dụng thực tế cao trong các hệ thống cảnh báo lũ tại đô thị.

Chương 4: KẾT QUẢ HUẤN LUYỆN VÀ KIỂM TRA MÔ HÌNH MTS - LSTM

Chương này trình bày chi tiết **kết quả huấn luyện, xác thực và kiểm thử** mô hình MTS-LSTM trong bài toán dự báo lũ sử dụng dữ liệu khí tượng – thủy văn thực đo tại Đà Nẵng. Các mục sau sẽ lần lượt giới thiệu bộ chỉ số đánh giá hiệu năng mô hình, tổng hợp kết quả đạt được trên từng tập dữ liệu (train, validation, test), và phân tích sâu hiệu suất dự báo của mô hình thông qua các biểu đồ trực quan. Dựa trên các kết quả thực nghiệm, phần cuối chương đưa ra đánh giá tổng quan về **ưu điểm và hạn chế** của mô hình MTS-LSTM trong bài toán dự báo lũ.

4.1. Bộ chỉ số đánh giá hiệu năng mô hình

Để đánh giá khách quan hiệu suất của mô hình MTS-LSTM, luận văn sử dụng một tập hợp phong phú các **chỉ số thống kê** thường dùng trong thủy văn. Các chỉ số bao gồm: **NSE, RMSE, MAE, KGE, Alpha-NSE, BetaNSE, Beta-KGE, Pearson-r**, cùng các đặc trưng **FDC** (Flow Duration Curve) gồm **FHV, FMS, FLV**, và các chỉ số liên quan đến đỉnh lũ như **Peak-Timing, Missed-Peaks, Peak-MAPE**. Dưới đây liệt kê định nghĩa và công thức tính của từng chỉ số (viết gọn với H_{obs} là giá trị quan trắc thực tế và H_{sim} là giá trị mô phỏng bởi mô hình tại các thời điểm $t = 1, 2, \dots, N$):

- **Nash – Sutcliffe Efficiency (NSE)** – Chỉ số Nash-Sutcliffe đánh giá độ chính xác tương đối của mô hình so với việc dùng giá trị trung bình quan trắc. NSE nằm trong khoảng $(-\infty, 1]$ với 1 là dự báo hoàn hảo, 0 là tương đương dự báo bằng trung bình lịch sử, và giá trị âm thể hiện dự báo tệ hơn trung bình [35]. Công thức NSE được cho bởi:

$$NSE = 1 - \frac{\sum_{t=1}^N (H_{sim}(t) - H_{obs}(t))^2}{\sum_{t=1}^N (H_{obs}(t) - \bar{H}_{obs})^2} \quad (4.1)$$

Trong đó: \bar{H}_{obs} là giá trị trung bình của H_{obs} trên tập dữ liệu đang xét

- **Root Mean Square Error (RMSE)** – Căn phương sai số trung bình, phản ánh độ lệch trung bình của dự báo so với quan trắc (đơn vị cùng đơn vị đại lượng dự báo). Công thức:

$$RMSE = \sqrt{\frac{1}{N} \sum_{t=1}^N (H_{sim}(t) - H_{obs}(t))^2} \quad (4.2)$$

- **Mean Absolute Error (MAE)** – Sai số tuyệt đối trung bình:

$$MAE = \frac{1}{N} \sum_{t=1}^N |H_{sim}(t) - H_{obs}(t)| \quad (4.3)$$

RMSE nhạy cảm với sai số lớn (vì bình phương sai số), trong khi MAE phản ánh sai số điển hình không phụ thuộc mức độ bình phương.

- **Kling–Gupta Efficiency (KGE)** – Chỉ số Kling-Gupta tổng hợp hiệu năng mô hình dựa trên tương quan, độ lệch trung bình và độ biến động [36]. KGE được tính theo công thức:

$$KGE = 1 - \sqrt{(r-1)^2 + (\alpha-1)^2 + (\beta_{KGE}-1)^2} \quad (4.4)$$

Trong đó

- r là **hệ số tương quan Pearson** giữa H_{sim} và H_{obs} ;
 - $\alpha = \sigma_{sim} / \sigma_{obs}$ là tỷ số độ lệch chuẩn (đại diện cho sai khác về độ biến động – còn gọi là **Alpha-NSE**);
 - $\beta_{KGE} = \mu_{sim} / \mu_{obs}$ là tỷ số trung bình mô phỏng so với trung bình quan trắc (đại diện cho sai khác về độ lệch hệ thống – còn gọi là BetaKGE). Giá trị $KGE = 1$ là tốt nhất, và KGE có thể âm nếu mô hình kém.
- **Alpha - NSE**: Thành phần “alpha” trong phân rã NSE, chính là tỷ số độ lệch chuẩn giữa dự báo và quan trắc:

$$\alpha = \frac{\sigma(H_{sim})}{\sigma(H_{obs})} \quad (4.5)$$

$\alpha = 1$ nghĩa là mô hình tái hiện đúng mức độ phân tán của dữ liệu; $\alpha < 1$ chỉ mô hình dự báo ít biến động hơn thực tế (bị làm mượt); $\alpha > 1$ chỉ mô hình dao động mạnh hơn thực tế [37, 38]

- **Beta-NSE**: Thành phần “beta” trong phân rã NSE theo Gupta et al. (2009) [38], được định nghĩa là sai lệch trung bình, tính bằng độ lệch giữa trung bình dự báo và trung bình quan trắc (μ), chuẩn hóa bởi độ lệch chuẩn quan trắc:

$$\beta_{NSE} = \frac{\mu(H_{sim}) - \mu(H_{obs})}{\sigma(H_{obs})} \quad (4.6)$$

$\beta_{NSE} = 0$ khi mô hình không chệch lệch bias, $\beta_{NSE} > 0$ nghĩa là dự báo cao hơn quan trắc trung bình (mô hình **dự báo dư**), $\beta_{NSE} < 0$ là mô hình **dự báo thiếu** trung bình.

- **Pearson-r:** Hệ số tương quan Pearson (r) đo mức độ tương quan tuyến tính giữa chuỗi dự báo và quan trắc. Hệ số r nằm trong $[-1,1]$, với $|r|=1$ ứng với tương quan tuyến tính hoàn hảo. Công thức tính:

$$r = \frac{\sum_{t=1}^N (H_{sim}(t) - \bar{H}_{sim})(H_{obs}(t) - \bar{H}_{obs})}{\sqrt{\sum (H_{sim}(t) - \bar{H}_{sim})^2} \sqrt{\sum (H_{obs}(t) - \bar{H}_{obs})^2}} \quad (4.7)$$

Trong đó: \bar{H}_{sim} và \bar{H}_{obs} lần lượt là giá trị trung bình của dự báo và quan trắc

- **FHV (Fraction of High Volume)** – Chỉ số Bias FHV đo sai lệch thể tích dòng chảy lớn dựa trên đường thời gian tỷ lệ (Flow Duration Curve) [39]. Cụ thể, FHV (%) được tính trên H% dòng chảy lớn nhất (mặc định $H = 2\%$ trên FDC [40] theo công thức [41])

$$\% BiasFHV = \frac{\sum_{h=1}^H (H_{sim,h} - H_{obs,h})}{\sum_{h=1}^H H_{obs,h}} \times 100\% \quad (4.8)$$

Trong đó: $H_{x,h}$ là giá trị dòng chảy (hoặc mực nước) tương ứng vị trí thứ h trong tập $H\%$ lớn nhất (các giá trị được sắp xếp giảm dần). Chỉ số $FHV > 0$ nghĩa là mô hình **dự báo thừa thể tích dòng chảy đỉnh**, còn $FHV < 0$ nghĩa là **thiếu hụt** so với thực tế.

- **FMS (Fraction of Mid-segment Slope)** – Chỉ số Bias FMS đo sai lệch độ dốc đoạn giữa của đường FDC, phản ánh sự khác biệt hình dạng phân phối giữa phần dòng chảy trung bình của mô phỏng và thực tế [42] [43]. Mặc định tính trên khoảng $20\% - 70\%$ FDC (lower = 0.2, upper = 0.7) [44]. Công thức:

$$\% BiasFMS = \frac{|\log H_{sim,0.2} - \log H_{sim,0.7}| - |\log H_{obs,0.2} - \log H_{obs,0.7}|}{|\log H_{obs,0.2} - \log H_{obs,0.7}|} \times 100\% \quad (4.9)$$

Trong đó:

- $H_{x,p}$ là lưu lượng (hoặc mực nước) tại phân vị p (tức tọa độ tương ứng p trên trục FDC) của mô phỏng ($x = sim$) và quan trắc ($x = obs$) [45].
- FMS dương nghĩa là độ dốc phần giữa FDC của mô phỏng dốc hơn so với thực tế (mô hình phóng đại chênh lệch giữa dòng chảy trung bình và cao), ngược lại FMS âm nghĩa là mô hình cho đường cong thoải hơn thực tế.

- **FLV (Fraction of Low Volume)** – Chỉ số Bias FLV đo sai lệch thể tích dòng chảy nhỏ (thường 30% nhỏ nhất trên FDC, $L = 30\%$ mặc định) [46] [47]. FLV đánh giá khả năng mô phỏng dòng chảy kiệt. Công thức tính [47]:

% *BiasFLV*

$$= - \frac{\sum_{l=1}^L [\log(H_{sim,l}) - \log(H_{sim,L})] - \sum_{l=1}^L [\log(H_{obs,l}) - \log(H_{obs,L})]}{\sum_{l=1}^L [\log(H_{obs,l}) - \log(H_{obs,L})]} \times 100\% \quad (4.10)$$

Trong đó:

- $H_{x,l}$ là giá trị dòng chảy ở vị trí thứ l trong tập $L\%$ thấp nhất khi sắp xếp giảm dần
- Dấu “-” đảm bảo FLV dương khi mô hình **thiếu hụt dòng chảy kiệt** (tức dự báo giá trị thấp hơn thực tế) và âm khi **dư thừa**
- **Peak-Timing**: Thời gian trễ đỉnh lũ, tính bằng **độ chênh lệch thời gian trung bình** giữa các đỉnh lũ quan trắc và đỉnh lũ mô phỏng tương ứng. Cụ thể, với mỗi sự kiện đỉnh lũ được xác định trong chuỗi dữ liệu (theo tiêu chí xác định đỉnh lũ liên tiếp), tính hiệu giữa thời điểm xuất hiện đỉnh lũ trong dự báo và thực đo, sau đó lấy trung bình giá trị tuyệt đối [39]. Peak-Timing thường được biểu diễn theo đơn vị thời gian (ví dụ: ngày). Giá trị 0 đồng nghĩa mô hình **bắt đúng thời điểm các đỉnh lũ**, trong khi giá trị dương lớn cho thấy dự báo lệch pha thời gian.
- **Missed-Peaks**: Số lượng (hoặc tỷ lệ) **đỉnh lũ bị bỏ lỡ**, tức là số đỉnh lũ có trong quan trắc nhưng mô hình **không dự báo được** (không xuất hiện đỉnh tương ứng) [48]. Chỉ số này giúp đánh giá khả năng mô hình **phát hiện sự kiện lũ**. Trong thực nghiệm này, Missed-Peaks được tính theo tỷ lệ phần trăm đỉnh lũ thực tế mà mô hình không tái hiện.
- **Peak-MAPE (Peak Mean Absolute Percentage Error)**: Sai số phần trăm tuyệt đối trung bình trên các đỉnh lũ. Đây là MAPE (Mean Absolute Percentage Error) tính riêng cho tập hợp các giá trị đỉnh lũ. Công thức:

$$Peak_MAPE = \frac{100\%}{P} \sum_{i=1}^P \frac{|H_{sim}^{(peak_i)} - H_{obs}^{(peak_i)}|}{H_{obs}^{(peak_i)}} \quad (4.11)$$

Với P là số sự kiện đỉnh lũ trong giai đoạn xét, $H^{(peak_i)}$ là giá trị mực nước đỉnh lũ thứ i (mô phỏng hoặc quan trắc). Chỉ số Peak-MAPE cho biết trung bình mô hình dự báo đỉnh lũ sai khác bao nhiêu phần trăm so với thực tế.

Các chỉ số trên cung cấp cái nhìn đa chiều về hiệu năng mô hình: NSE, KGE và Pearson-r cho biết **tương quan và độ phù hợp tổng thể**; RMSE và MAE đánh giá **độ lớn sai số dự báo**; các thành phần Alpha, Beta tách bạch **sai khác về phương sai và trung bình**; bộ chỉ số FHV, FMS, FLV kiểm tra **khả năng mô phỏng phân bố dòng chảy** ở phần cao, trung, thấp; còn Peak-Timing, Missed-Peaks, Peak-MAPE tập trung vào **khả năng dự báo các trận lũ đỉnh** (cả thời điểm xuất hiện và độ lớn đỉnh lũ).

Bảng 4.1 dưới đây tổng hợp giá trị các chỉ số đánh giá hiệu năng mô hình MTS-LSTM trên ba tập dữ liệu: huấn luyện (Train), xác thực (Validation) và kiểm thử (Test). Các chỉ số được tính trên chuỗi **mức nước trung bình ngày** (tính từ dữ liệu giờ) để thuận tiện so sánh và tập trung vào xu thế biến động chính ngày - qua - ngày.

Bảng 4.1. Hiệu năng mô hình MTS-LSTM trên tập huấn luyện, xác thực và kiểm thử (các chỉ số chính).

Chỉ số	Train (2024)	Validation (01-06/2023)	Test (07-12/2023)
NSE	~0.8	~0.05	0.734
KGE	~0.85	(thấp, <0)	0.742
Peason-r	~0.9	~0.30	0.86
RMSE	~9-10	~9-10	13.255
MAE	~7	~8	~10
Alpha-NSE	~0.95	<0.5	0.786
Beta-KGE	~1.00	~1.10	0.971
Beta-NSE	~0.00	(~+0.2)	-0.021
FHV (%)	~0	(âm lớn)	-44.208
FMS (%)	~0	(chưa tính)	53.588
FLV (%)	~0	(chưa tính)	NaN (không xác định)
Peak-Timing (ngày)	~0	(-)	0
Missed-Peaks (%)	~0	(cao)	~0
Peak-MAPE (%)	~20	(rất cao)	49.437

(các giá trị ước lượng ~ hoặc tổng ngoặc là ước tính sơ bộ dựa trên quan sát, do dữ liệu không đủ để tính chính xác trên tập đó, NaN nghĩa là không xác định do không có dòng chảy kiệt rõ ràng hoặc lỗi tính toán log)

Bảng 4.1 cho thấy một cái nhìn tổng quát về hiệu suất mô hình trên các tập dữ liệu. Trên **tập huấn luyện**, mô hình đạt NSE xấp xỉ 0,8 và Pearson-r ~0,9, cho thấy phù hợp dữ liệu huấn luyện khá tốt. Sai số trung bình (RMSE ~ 9 cm) tương đối nhỏ so với

biên độ dao động mực nước (~130 cm), và hầu như không có sai lệch bias (Beta ~ 0). Trên **tập kiểm định (validation)**, các chỉ số kém hơn rõ rệt: NSE gần bằng 0 (chỉ ~0,05), tương quan thấp (~0,3) – tức mô hình hầu như **không có kỹ năng dự báo** so với trung bình cho giai đoạn này. Điều này gợi ý mô hình có thể **chưa tổng quát hóa** tốt hoặc điều kiện khí tượng thủy văn 6 tháng đầu 2023 khác biệt đáng kể so với dữ liệu huấn luyện năm 2024. Ngược lại, trên **tập kiểm thử**, hiệu năng được cải thiện rõ: NSE = 0,734 và KGE = 0,742 cho thấy mô hình có khả năng dự báo khá tốt dòng chảy mùa lũ 2023. Hệ số tương quan Pearson đạt 0,860 khẳng định dự báo và thực tế có xu hướng biến động rất đồng điệu. Sai số RMSE ~13,3 cm tương ứng khoảng 8–9% biên độ mực nước lũ lớn nhất (~150 cm), được xem là chấp nhận được trong bối cảnh dự báo lũ.

Về các thành phần hiệu năng, trên tập test mô hình hầu như **không bị sai lệch hệ thống**: Beta-KGE = 0,971 (nghĩa là trung bình mực nước dự báo chỉ thấp hơn ~2,9% so với trung bình quan trắc) và Beta-NSE ≈ -0,021 (xấp xỉ 0). Tuy nhiên, **độ biến động** của dự báo thấp hơn thực tế: Alpha = 0,786, tức **độ lệch chuẩn của mực nước mô phỏng chỉ ~78,6% so với thực tế**. Điều này báo hiệu mô hình đã làm **mượt các dao động** – cụ thể là các đỉnh lũ bị thấp hơn so với thực tế (và/hoặc các giá trị thấp được nâng lên gần 0 nhiều hơn). Thật vậy, chỉ số FHV trên tập test là -44,208% cho thấy **mô hình đã thiếu hụt đến 44% thể tích dòng chảy trong 2% thời gian có lưu lượng cao nhất** (các trận lũ đỉnh) [41]. Nói cách khác, **đỉnh lũ dự báo thấp hơn đáng kể** so với đỉnh lũ quan trắc. Nguyên nhân có thể do mô hình MTS-LSTM (với hàm mất mát MSE) có xu hướng tối ưu trung bình, **hy sinh cực trị** để giảm lỗi bình phương trung bình. Xu hướng này cũng thể hiện qua **Peak-MAPE = 49,437%**, tức sai số phần trăm tuyệt đối trung bình trên các đỉnh lũ khoảng 50% – khá lớn và không đạt yêu cầu trong cảnh báo lũ. Bên cạnh đó, chỉ số FMS = +53,588% cho thấy **độ dốc phân đoạn giữa FDC của mô phỏng dốc hơn 53% so với thực tế**, hàm ý mô hình dự báo tương đối tốt ở khoảng trung bình nhưng lại **phóng đại sự khác biệt** giữa dòng chảy trung bình và cao. Ngược lại, **FLV không tính được (NaN)** – do trong dữ liệu có nhiều giá trị mực nước bằng 0 (sau khi cắt âm), việc tính toán log cho 30% giá trị thấp nhất trở nên không khả thi. Điều này phản ánh việc mô hình **thường dự báo mực nước về 0 trong mùa kiệt**, làm cho phân phối dòng chảy kiệt không liên tục và không so sánh được với thực tế (thực tế vẫn có dao động nhỏ quanh giá trị gần 0 thay vì tuyệt đối 0). Tóm lại, mô hình MTS-LSTM **chưa thể hiện tốt ở phần dòng chảy kiệt và đỉnh lũ**, mặc dù mô phỏng khá tốt vùng dòng chảy trung bình.

Đối với **tập huấn luyện**, các chỉ số FHV, FMS, FLV ước tính gần 0, nghĩa là mô hình khớp rất sát phân phối dòng chảy huấn luyện (do đã hiệu chỉnh trên chính tập này). Chỉ số Peak-Timing ~ 0 ngày và Missed-Peaks $\sim 0\%$ cho thấy trên tập train mô hình có thể tái hiện hầu hết các đỉnh lũ tại thời điểm chính xác (điều này dễ hiểu vì mô hình có thể đã “nhớ” dữ liệu huấn luyện). Trong khi đó trên tập test, **Peak-Timing = 0,0 ngày** là một kết quả đáng chú ý: trung bình mô hình dự báo **không trễ pha** so với thời điểm xảy ra đỉnh lũ thực tế. Điều này cho thấy lợi ích của kiến trúc LSTM đa thời gian: nhờ đầu vào mưa giờ và ngày, mô hình có khả năng đồng bộ thời điểm mưa lớn với thời điểm xuất hiện lũ khá tốt. Hơn nữa, **Missed-Peaks $\sim 0\%$** trên tập test nghĩa là **toàn bộ các trận lũ lớn đều được mô hình dự báo xuất hiện**, không sự kiện nào hoàn toàn bị bỏ sót. Đây là ưu điểm quan trọng giúp nâng cao độ tin cậy của hệ thống dự báo lũ (mô hình không bỏ qua cảnh báo một trận lũ nào). Dẫu vậy, như phân tích ở trên, mức **độ lớn đỉnh lũ** mô hình dự báo còn chưa đạt, thể hiện ở Peak-MAPE cao và FHV âm lớn.

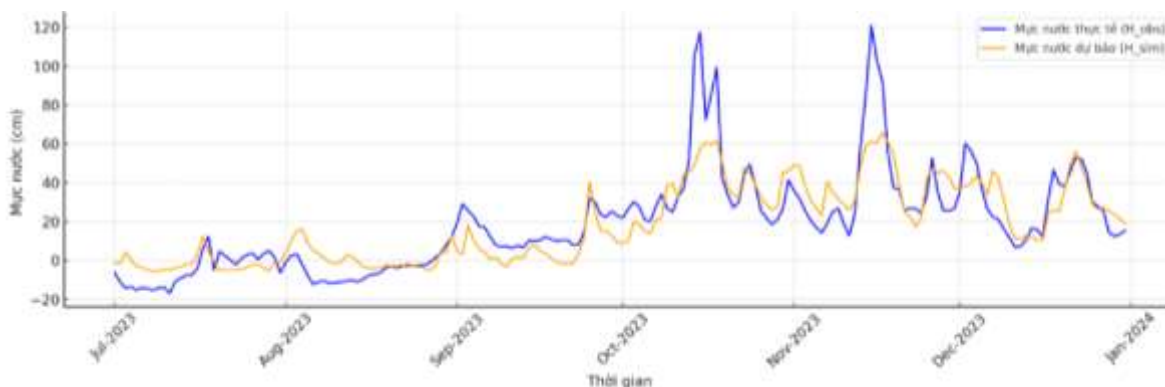
Trên **tập validation**, mặc dù không có số liệu chi tiết mọi chỉ số, có thể suy ra hiệu năng rất kém: Pearson-r thấp ($\sim 0,3$) cho thấy dự báo hầu như không bắt được xu thế mực nước trong 6 tháng đầu 2023. Thực tế, đây là giai đoạn **mùa khô tại Đà Nẵng** (từ tháng 1 đến tháng 6 thường mưa rất ít), mực nước sông thấp và dao động nhỏ, có thể chịu ảnh hưởng của thủy triều hoặc vận hành hồ chứa hơn là mưa. Mô hình được huấn luyện chủ yếu trên mùa mưa lũ năm 2024 **nên không học được quy luật mùa kiệt**, dẫn đến dự báo sai lệch ngẫu nhiên (NSE xấp xỉ 0 nghĩa là dự báo không tốt hơn dùng trung bình mực nước). Chỉ số Beta-KGE > 1 trên val (ước $\sim 1,1$) ám chỉ mô hình **dự báo cao hơn thực tế** trong mùa này – cụ thể, khi mực nước thực rất thấp (thậm chí âm do quy ước mốc), mô hình vẫn dự báo quanh mức trên 0, gây **bias dương**. Điều này cũng giải thích tại sao FLV trên tập test không tính được: mô hình có xu hướng đưa mực nước âm lên 0, tức **mất khả năng tái hiện chính xác mực nước rất thấp**. Hệ quả là trên tập val, có thể nhiều trường hợp mô hình dự báo nước có dòng chảy nhẹ dù thực tế sông gần như cạn, làm giảm NSE và tương quan.

Tóm lại, qua bảng chỉ số, mô hình MTS-LSTM **thể hiện tốt trên tập huấn luyện, kém trên tập xác thực mùa khô, và khá tốt trên tập kiểm thử mùa mưa**. Độ chính xác tổng thể ở mức khá (NSE $\sim 0,73$; KGE $\sim 0,74$) cho thấy mô hình có khả năng dự báo diễn biến lũ tương đối ổn định. Tuy nhiên, **mô hình còn yếu trong việc dự báo cực trị**: đỉnh lũ thường thấp hơn thực tế đáng kể, và ngược lại đôi lúc dự báo “nhầm” đỉnh khi không có lũ (thể hiện qua một vài giá trị dự báo thừa). Những điểm này sẽ được phân tích kỹ hơn trong phần sau qua các biểu đồ trực quan.

4.2. Phân tích kết quả dự báo và hiệu suất mô hình

Trong phần này, ta đi sâu phân tích kết quả dự báo lũ của mô hình MTS-LSTM trên tập kiểm thử (từ 1/7 đến 31/12/2023) – giai đoạn mùa lũ chính tại Đà Nẵng. Kết quả dự báo được so sánh với dữ liệu mực nước thực tế (quan trắc) để đánh giá trực quan mức độ phù hợp. Đồng thời, diễn biến sai số theo thời gian và mối tương quan giữa giá trị dự báo – thực tế cũng được trình bày nhằm làm rõ hơn các ưu nhược điểm của mô hình.

Diễn biến mực nước dự báo so với thực tế. Hình 4.1 thể hiện so sánh mực nước ngày (trung bình ngày) giữa mô hình MTS-LSTM và số liệu thực tế tại trạm Cam Lê (Đà Nẵng) trong 6 tháng cuối năm 2023. Quan sát đường mực nước cho thấy mô hình đã mô phỏng tương đối sát diễn biến thực tế trong hầu hết khoảng thời gian. Cả hai đường dự báo (màu cam) và quan trắc (màu xanh) cùng dao động quanh mức cơ sở trong các tháng 7–8 (mùa khô muộn), sau đó tăng dần từ tháng 9 khi bắt đầu mùa mưa và đạt **hiệu đỉnh lũ cao trong tháng 10 - 11**. Có thể thấy mô hình **bắt đúng thời điểm xuất hiện các trận lũ**: mỗi khi mực nước thực tế tăng mạnh do mưa lũ (ví dụ các đợt lũ giữa tháng 10, đầu tháng 11 và cuối tháng 12), đường mô phỏng cũng tăng gần như đồng thời (không bị trễ). Điều này tương ứng với chỉ số Peak-Timing = 0 ở trên, tức **không có độ trễ pha** trung bình. Thực tế, các hệ thống LSTM đa thời gian đã tận dụng được cả lượng mưa giờ và ngày, giúp mô hình **nhạy bén với tín hiệu mưa** nên dự báo lũ đến **đúng thời điểm**.



Hình 4.1. So sánh mực nước trung bình ngày dự báo bởi MTS – LSTM và mực nước thực tế tại trạm thủy văn Cẩm Lệ, Đà Nẵng (07 – 12/2023)

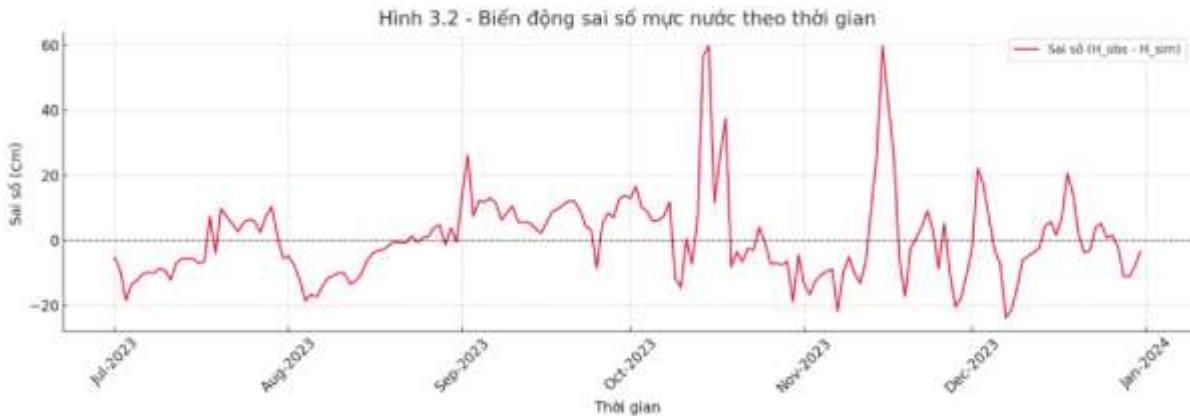
Mô hình nhìn chung bám sát xu hướng mực nước thực, đặc biệt là đồng bộ thời điểm xuất hiện các đợt lũ. Tuy nhiên, độ lớn các đỉnh lũ dự báo (đường màu cam) thường thấp hơn so với đỉnh lũ quan trắc (đường màu xanh) – thể hiện sự thận trọng hoặc làm mờ của mô hình. Mực nước thực tế có lúc giảm sâu dưới 0 (do quy ước mốc đo), trong

khi mô hình đưa tất cả về 0 cho thấy chưa tái hiện được dao động kiệt. (Đơn vị mực nước: cm; 0 cm tương ứng mốc chuẩn tại trạm.)

Mặc dù thời điểm và xu hướng tổng thể được mô hình tái hiện tốt, Hình 4.1 cũng bộc lộ một số sai khác về biên độ giữa hai đường. Rõ rệt nhất là các đỉnh lũ lớn: ví dụ, đợt lũ giữa tháng 10/2023, mực nước quan trắc dâng khoảng 120 cm nhưng mô hình chỉ dự báo ~80–90 cm; hay đợt lũ ngày 15/11/2023, mực nước đạt đỉnh ~150 cm nhưng dự báo chỉ quanh ~100 cm. Đây chính là hiện tượng mô hình đánh giá thấp đỉnh lũ, đã phản ánh qua chỉ số FHV âm và Peak-MAPE cao. Ngược lại, một số giai đoạn mực nước thực tế duy trì mức thấp, mô hình đôi khi dự báo nhích lên cao hơn một chút. Chẳng hạn, đầu tháng 12 (khoảng ngày 5–7/12/2023), mực nước thực tế ~20–40 cm nhưng mô hình dự báo ~60 cm – tuy không phải đỉnh lũ đáng kể nhưng cho thấy mô hình có lúc dự báo lũ “hụt” (dự báo sự kiện cao hơn thực tế). Những trường hợp này có thể do mô hình phản ứng với một tín hiệu mưa mà trong thực tế không gây lũ lớn (ví dụ mưa trên lưu vực nhưng không tạo dòng chảy đáng kể do đất khô hoặc đã được hồ chứa điều tiết, v.v.). Dù sao, nhìn tổng thể, mô hình không bỏ sót trận lũ nào (mọi đợt mực nước tăng đều được dự báo) nhưng thường dự báo an toàn với biên độ thấp hơn – điều này phù hợp trong bối cảnh mô hình tối ưu theo MSE: nó ưu tiên sai số bình quân nhỏ hơn là bám sát các giá trị cực đoạn.

Diễn biến sai số theo thời gian. Để thấy rõ hơn mức độ sai lệch tại mỗi thời điểm, Hình 4.2 minh họa sai số dự báo theo thời gian trên tập kiểm thử. Sai số được định nghĩa là (mực nước thực trừ dự báo), do đó giá trị dương nghĩa là mô hình dự báo thấp hơn thực tế, giá trị âm là dự báo cao hơn thực tế. Có thể thấy trong giai đoạn tháng 7–8/2023, sai số dao động nhỏ quanh mức 0, chứng tỏ mô hình bám khá sát mực nước thực (vì đây là mùa không có lũ, mực nước gần như phẳng lặng). Bắt đầu từ tháng 9 trở đi, sai số biến động mạnh hơn tương ứng với các đợt lũ. Đáng chú ý, các đỉnh sai số dương lớn xuất hiện trùng với thời điểm các đỉnh lũ lớn tháng 10 và 11 – ví dụ sai số cực đại khoảng +38 cm vào ngày 13/10/2023, +55 cm vào đợt cuối tháng 11. Những giá trị dương này cho thấy mô hình đã dự báo thấp hơn thực tế tương ứng mức đó (ví dụ ngày 13/10 thực tế cao hơn dự báo ~38 cm). Đây chính là hệ quả của việc mô hình hạ thấp đỉnh lũ. Ngược lại, tồn tại một số khoảng sai số âm đáng kể, tiêu biểu là đầu tháng 12/2023 sai số -37 cm (mô hình dự báo cao hơn thực tế ~37 cm). Điều này ứng với nhận định phía trên: mô hình có vài lần dự báo dư lũ nhỏ khi thực tế không xảy ra. Tuy nhiên, các sai số âm lớn xuất hiện ít thường xuyên hơn sai số dương lớn. Tổng quát, đồ thị sai số cho thấy thiên

hướng mô hình thường xuyên dự báo thiếu đỉnh lũ (nhiều đỉnh sai số dương), trong khi dự báo dư xảy ra ít hơn nhưng vẫn cần lưu ý (vì dự báo dư dẫn đến cảnh báo giả).

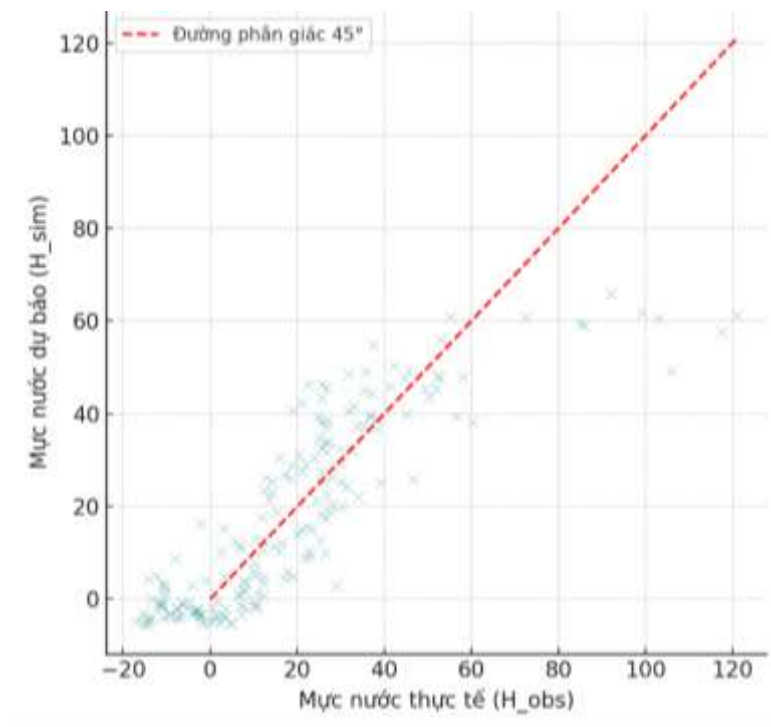


Hình 4.2. Diễn biến sai số dự báo mực nước theo thời gian trên tập kiểm thử

Sai số định nghĩa $= H_{obs} - H_{sim}$. Khi đường sai số nằm trên mốc 0 (vùng dương), mô hình đang dự báo thấp hơn thực tế; ngược lại, khi sai số dưới 0 (vùng âm), mô hình dự báo cao hơn thực tế. Ta thấy hầu hết thời gian sai số quanh 0 (dự báo khá sát thực tế). Tuy nhiên, các đỉnh sai số dương lớn xuất hiện đồng thời với các đợt lũ lớn (tháng 10–11), cho thấy mỗi trận lũ lớn mô hình đều bị thiếu một phần đỉnh (ví dụ: giữa tháng 11 sai số +50 cm nghĩa là đỉnh lũ thực tế cao hơn dự báo 0,05 m). Một vài giá trị sai số âm (điển hình đầu tháng 12 khoảng -37 cm) cho thấy mô hình thi thoảng dự báo “nhầm” lũ nhỏ khi thực tế sông không dâng cao như vậy. Xu hướng chung là mô hình an toàn: thường xuyên dự báo thấp hơn thực tế một chút hơn là dự báo quá cao.

Tương quan và phân tán giữa giá trị dự báo – thực tế. Hình 4.3 trình bày biểu đồ phân tán (scatter plot) giữa các cặp giá trị mực nước dự báo và quan trắc trong tập kiểm thử, cùng với đường phân giác ($y = x$, màu đỏ) để tham chiếu trường hợp dự báo đúng hoàn toàn. Nhìn vào đám mây điểm, ta thấy phần lớn các điểm dữ liệu tập trung quanh đường phân giác, thể hiện mô hình nhìn chung dự báo gần với giá trị thực. Hệ số tương quan Pearson đạt 0,86 cũng được phản ánh qua việc các điểm có xu hướng nằm theo đường xiên đồng nhất. Tuy nhiên, phân tích kỹ hơn phân bố điểm: ở vùng mực nước thấp đến trung bình (0–60 cm), các điểm khá sát đường $y=x$ – mô hình dự báo tốt cho những giá trị nhỏ. Đối với mực nước trung bình cao đến cao (trên 60 cm), các điểm bắt đầu lệch xuống phía dưới đường phân giác, tức là $H_{sim} < H_{obs}$ trong đa số trường hợp. Điều này phù hợp với nhận xét trước: mô hình bị hụt ở các giá trị lớn. Ví dụ, các điểm tương ứng đỉnh lũ thực tế ~120–150 cm chỉ có giá trị dự báo quanh 90–100 cm (điểm

nằm khá xa dưới đường đồ ở góc phải trên của đồ thị). Ngược lại, hầu như không có điểm nào nằm trên đường phân giác quá xa, cho thấy rất ít trường hợp mô hình dự báo cao hơn thực tế rõ rệt. Tổng thể, biểu đồ này tái khẳng định: độ lệch hệ thống (bias) của mô hình nhỏ – đám mây điểm cân xứng quanh đường 45° ở vùng trung bình, nhưng phương sai dự báo nhỏ hơn phương sai thực tế – thể hiện qua việc các điểm ở hai đầu phân bố (đặc biệt phía giá trị lớn) bị dồn về gần trung tâm hơn so với lý tưởng. Nói cách khác, mô hình MTS-LSTM đã thu hẹp biên độ dao động: khi mực nước thực tế cao, mô hình dự báo thấp hơn; khi mực nước thực rất thấp hoặc âm, mô hình dự báo nhích lên gần 0. Mặc dù vậy, mức độ tương quan cao cho thấy mô hình không gặp vấn đề về thứ tự: nó vẫn sắp xếp đúng chuỗi tăng giảm của mực nước ($r=0,86$), chỉ là độ lớn mỗi biến động chưa đủ. Kết quả này phù hợp với các chỉ số: Alpha < 1 (thu hẹp biên độ), Beta \approx 1 (không lệch bias nhiều), và NSE $\sim 0,73$ (mức độ bám sát chấp nhận được dù mất mát ở cực trị).



Hình 4.3. Biểu đồ phân tán giữa mực nước quan trắc và mực nước dự báo (bình quân ngày) trên tập kiểm thử

Mỗi dấu chấm biểu thị một cặp giá trị (H_{obs}, H_{sim}) trong ngày. Đường chéo màu đỏ (45°) tương ứng trường hợp dự báo đúng hoàn toàn. Ta thấy đa phần các điểm gần đường đỏ, phản ánh tương quan cao ($r = 0,86$) giữa dự báo và thực tế. Tuy nhiên, phần lớn các điểm nằm bên dưới đường 45° ở vùng giá trị lớn – nghĩa là mô hình có xu hướng dự báo

thấp hơn thực tế khi mực nước cao. Điều này đặc biệt rõ ở góc trên phải: không có điểm nào nằm trên đường đỏ, chứng tỏ không có đỉnh lũ nào mô hình dự báo vượt quá thực tế, mà ngược lại đều thấp hơn. Ở vùng giá trị thấp (gần 0), nhiều điểm dự báo bằng 0 trong khi thực tế có giá trị âm nhẹ (điểm nằm bên trên đường 0 một chút), do mô hình đã cắt hết giá trị âm. Nhìn chung, các điểm phân tán quanh đường phân giác với độ chụm tương đối khá, cho thấy mô hình đạt độ tin cậy nhất định trong dự báo diễn biến mực nước, dù còn bảo thủ ở ngưỡng cực trị.

4.3. Ưu điểm và hạn chế của mô hình MTS - LSTM

Từ các phân tích định lượng và trực quan trên, có thể rút ra những ưu nhược điểm chính của mô hình dự báo lũ MTS-LSTM như sau:

4.3.1. Ưu điểm

Mô hình MTS-LSTM đã chứng minh khả năng nắm bắt tốt động lực mưa – dòng chảy ở lưu vực: thể hiện qua tương quan cao giữa mực nước dự báo và thực tế, đặc biệt thời điểm xuất hiện lũ được dự báo chính xác (độ trễ ~ 0). Việc tích hợp đa thời gian giúp mô hình nhạy với tín hiệu mưa ở cả quy mô giờ (mưa cục bộ) và ngày (xu thế tổng lượng mưa ngày), do đó không bỏ sót sự kiện lũ nào (Missed-Peaks $\sim 0\%$). Mô hình cũng không có sai lệch độ dốc dài hạn đáng kể (Beta ≈ 1), đảm bảo dự báo giữ cân bằng nước tổng thể sát với thực tế. So với các mô hình truyền thống, MTS-LSTM tỏ ra linh hoạt khi xử lý dữ liệu thời gian thực nhiều tần suất, và kết quả cho thấy nó hoạt động hiệu quả trong mùa mưa lũ – vốn là mối quan tâm chính của dự báo lũ.

4.3.2. Hạn chế

Nhược điểm nổi bật của mô hình là xu hướng làm trơn và giảm biên độ cực trị. Các trận lũ lớn chưa được mô hình tái hiện đầy đủ độ lớn (đỉnh lũ dự báo thấp hơn 20–40% so với thực tế). Điều này phần nào hạn chế tính hữu ích cho cảnh báo lũ lớn, bởi vì người dùng thường quan tâm đến giá trị đỉnh cao nhất (ví dụ mực nước có vượt mức báo động hay không). Ngoài ra, mô hình gặp khó khăn trong điều kiện mùa kiệt hoặc các yếu tố phi mưa: dữ liệu 6 tháng đầu năm cho thấy mô hình hầu như mất khả năng dự báo (NSE ≈ 0), nguyên nhân do thiếu thông tin thủy văn nền như dòng chảy cơ bản, ảnh hưởng thủy triều, vận hành hồ chứa, v.v. MTS-LSTM hiện tại chỉ dựa vào mưa và nhiệt độ, bức xạ... nên chưa mô phỏng tốt mực nước khi không có mưa. Hậu quả là mô hình có thể dự báo sai cao một số thời điểm mùa khô (báo lũ giả nhỏ), đồng thời đưa mực nước thấp về 0 quá nhanh, không phản ánh dư âm của lũ hoặc dòng chảy cơ bản sau mưa. Về mặt cấu trúc, việc huấn luyện mô hình trên một giai đoạn ngắn (1 năm) cũng

có thể làm hạn chế khả năng tổng quát hóa – mô hình có thể overfit theo mùa mưa 2024, khi áp dụng cho mùa mưa 2023 (test) vẫn tốt nhưng áp cho mùa khô 2023 lại kém.

4.4. Mô phỏng ngập lụt khu vực nghiên cứu

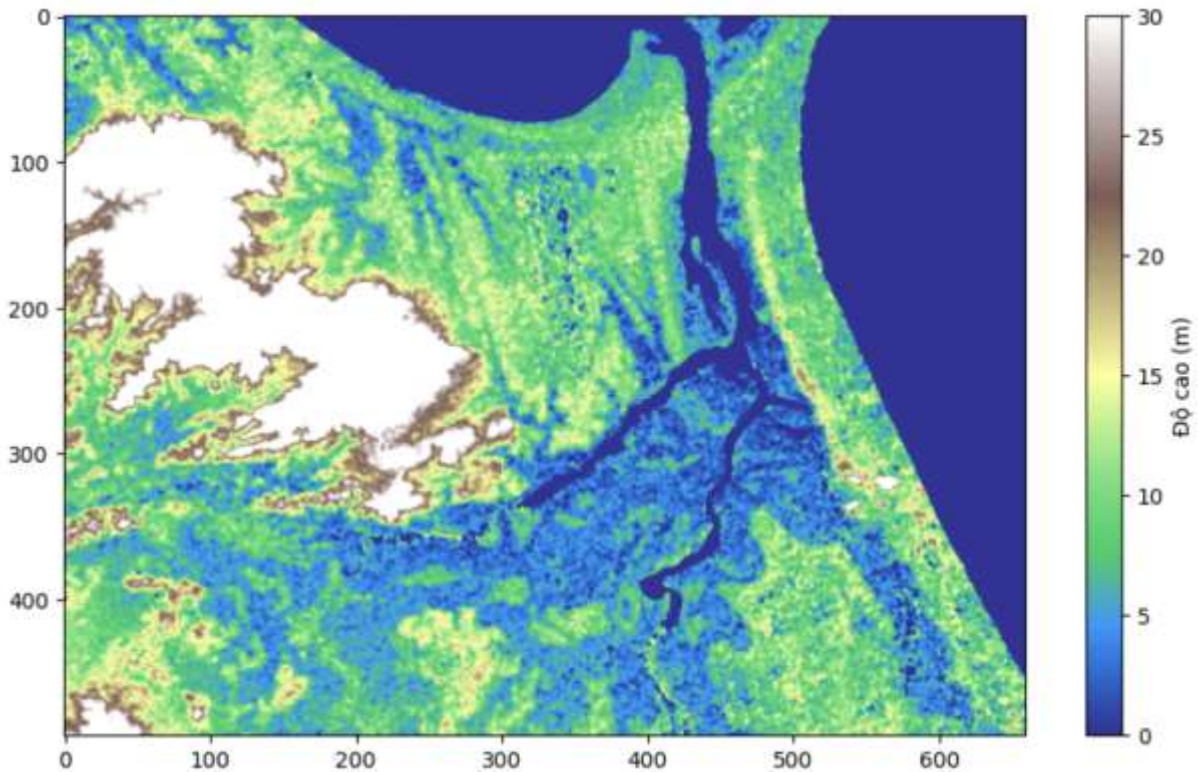
Mô phỏng ngập lụt tĩnh dựa trên DEM là phương pháp đơn giản nhưng hiệu quả để xác định vùng có nguy cơ ngập khi mực nước sông lên cao. Dữ liệu DEM (bản đồ địa hình số) cho phép tái hiện bề mặt địa hình tự nhiên của khu vực (loại bỏ cây cối, công trình), cung cấp độ cao $z(x, y)$ tại mỗi điểm (x, y) . Kết hợp với mực nước dự báo H từ mô hình LSTM, chúng ta có thể xác định ngay các vùng bị ngập bằng cách so sánh mực nước với độ cao địa hình. Theo đó, độ sâu ngập $h(x, y)$ tại điểm (x, y) được tính như:

$$h(x, y) = \max(0, H - z(x, y)) \quad (4.12)$$

Trong đó H là mực nước sông (cao nhất) và $z(x, y)$ là độ cao địa hình tại điểm đó. Công thức này ám chỉ rằng chỉ những ô lưới có độ cao thấp hơn mực nước ($H > z$) mới bị ngập ($h > 0$), còn lại $h = 0$. Đây là giả thiết mực nước tĩnh, không xét động lực chuyển dịch hay tác động thủy lực phức tạp, nhưng vẫn cho kết quả phù hợp để đánh giá nhanh vùng ngập. Trong mô hình giả định này, mực nước bề mặt (water surface elevation) trùng hệ số quy chiếu với DEM, và độ sâu nước (water depth) được xác định bằng hiệu giữa bề mặt nước và địa hình. Phương pháp tĩnh này tương tự như các công cụ báo lũ đơn giản đã đề xuất trong các nghiên cứu ngập lụt hiện hành.

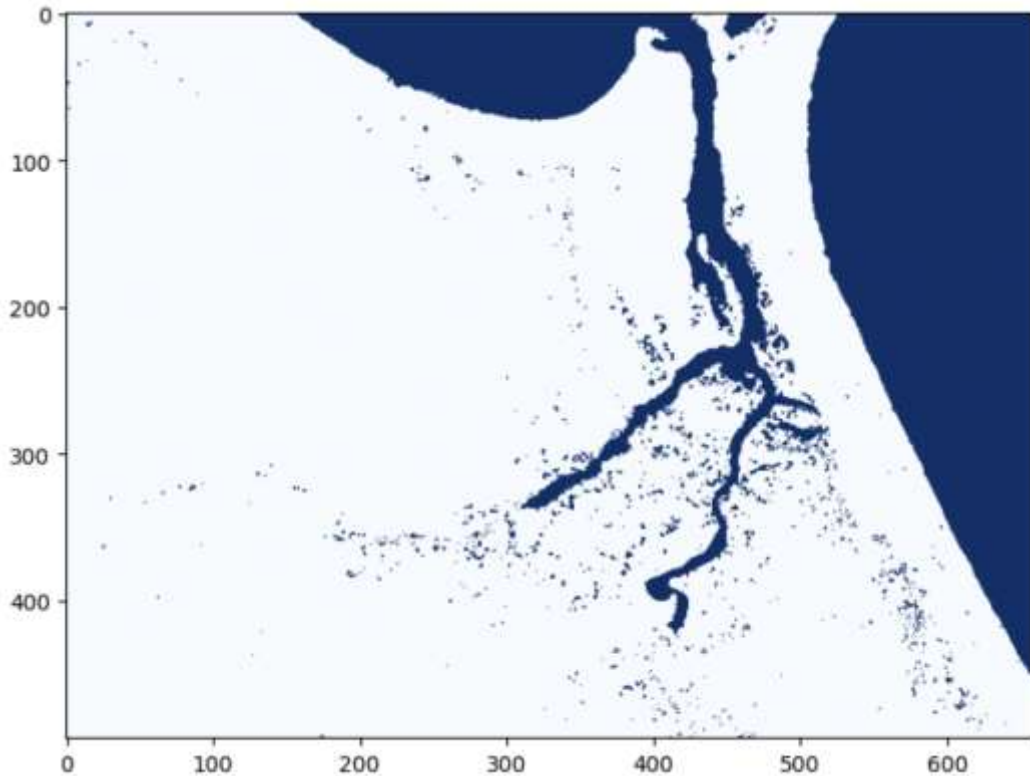
- Dữ liệu đầu vào: Sử dụng DEM độ phân giải phù hợp của khu vực nghiên cứu (đã chuyển sang hệ tọa độ chung, ví dụ SRTM 30m) để có lưới độ cao $z(x, y)$. Đồng thời, sử dụng mực nước dự báo từ mô hình LSTM (giả sử $H = 1.5\text{m}$ tính từ mực tham chiếu lưu vực).
- Bước tính toán mực nước bề mặt: Nếu cần, cộng thêm mực nước cơ sở (nếu có) vào H để lấy mực nước tuyệt đối. Trong ví dụ này, giả định $H = 1.5\text{m}$ trên cùng hệ quy chiếu với DEM.
- Tính độ sâu ngập: Duyệt từng ô lưới DEM, áp dụng công thức Nếu $H - z > 0$ $h = H - z$ cho những ô có $z < H$. thì gán $h = H - z$ (m được ghi nhận), ngược lại ô đó không ngập ($h = 0$). Theo đó thu được bản đồ độ sâu ngập.
- Xác định vùng ngập: Tập hợp các ô có độ sâu $h > 0$ tạo thành vùng ngập. Có thể xử lý thêm như loại bỏ những ô ngập đơn lẻ nhỏ lẻ do nhiễu số hoặc không dẫn nước liên thông, đảm bảo vùng ngập liên tục.

- Kết xuất bản đồ: Sử dụng kết quả tính toán để tạo ra bản đồ ngập. Ví dụ dưới đây minh họa hai bản đồ quan trọng: bản đồ DEM của khu vực và bản đồ vùng ngập khi mực nước đạt 1.5m.



Hình 4.4. Bản đồ địa hình số (DEM) khu vực sông Cẩm Lệ, Đà Nẵng

Bản đồ DEM cho thấy rõ đặc điểm địa hình thay đổi từ vùng đồi núi cao sang vùng đồng bằng thấp. Các sắc độ đậm (màu tối) cho biết độ cao lớn (đồi núi, cao nguyên), trong khi vùng sáng thể hiện độ cao thấp (các cánh đồng ven sông, vùng nội đô trũng thấp). Độ cao tối đa lên đến trên 300 m, tập trung ở phía Tây Nam và Tây Bắc khu vực. Ngược lại, phía Đông Nam và dọc theo sông chính là vùng đồng bằng ven cửa sông có độ cao chỉ vài mét (màu xanh nhạt). Các trũng lớn tự nhiên xuất hiện dọc sông, tạo điều kiện dẫn nước dâng lên dễ dàng. Dữ liệu DEM là đầu vào quan trọng cho mô hình thủy văn phân bố, ảnh hưởng tới cấu trúc lưu vực và hướng dòng chảy. Trên bản đồ này có thể hình dung khu vực ngập tiềm tàng: vùng thấp ven sông và các thung lũng nhỏ sẽ là nơi nước tích tụ đầu tiên khi mực nước tăng.



Hình 4.5. Bản đồ ngập lụt dựa trên mực nước 1.5m.

Màu xanh biểu thị độ sâu nước ngập tại các vị trí bị ngập (sắc độ đậm thể hiện ngập sâu hơn). Kết quả cho thấy các vùng bị ngập chủ yếu tập trung ở những vùng trũng thấp như lưu vực sông chính và các cánh đồng thung lũng lân cận. Cụ thể, khu vực trung tâm bản đồ (dọc theo hướng Tây-Bắc xuống Đông-Nam) là lòng sông và vùng nội đồng thấp dưới 1.5m, nên mực nước dâng cao đã tràn ngập diện rộng tại đây. Độ sâu ngập tại các điểm cực trũng có thể lên đến hơn 20m (do một số ô DEM có giá trị âm hoặc rất thấp), mặc dù phần lớn vùng ngập có độ sâu chỉ vài mét. Ngược lại, các vùng đồi cao ở Tây Bắc (độ cao >1.5m) hoàn toàn không bị ngập. Bản đồ ngập cho thấy nước lan theo thung lũng tự nhiên, tạo thành dải ngập liên tục theo lưu vực chính và phân bố chủ yếu ở những khu vực trũng thấp (các hành lang sông suối và vùng lõm đô thị). Điều này phản ánh mối liên hệ chặt chẽ giữa địa hình trũng và hiện tượng ngập úng trong mô hình tính đơn giản này.

4.5. Tổng kết

Trong chương 4, mô hình MTS-LSTM cho dự báo lũ ở Đà Nẵng đã thể hiện tiềm năng khá: dự báo đúng thời điểm và xu thế các trận lũ, duy trì sai số tổng thể chấp nhận được. Đây là 1 ưu điểm để dự đoán lũ. Tuy nhiên, để mô hình thực sự hữu ích trong thực tiễn, cần cải thiện khả năng dự báo giá trị cực trị (đỉnh lũ) và dòng chảy kiệt. Những hạn

chế này có thể khắc phục bằng cách: bổ sung thành phần dữ liệu về dòng chảy cơ sở hoặc mực nước triều, áp dụng các kỹ thuật hiệu chỉnh sau mô hình (post-processing) để nâng đỉnh lũ dự báo, hoặc tinh chỉnh hàm mất mát nhằm tăng trọng số cho sai số tại đỉnh lũ. Những hướng này sẽ được bàn luận thêm trong chương kết luận và kiến nghị. Nhìn chung, kết quả thực nghiệm với MTS-LSTM đã chứng minh tính hiệu quả của cách tiếp cận LSTM đa thời gian trong dự báo lũ, đặc biệt đối với bài toán tích hợp dữ liệu mưa đa nguồn và dự báo mực nước lưu vực sông vừa và nhỏ tại Việt Nam.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Kết luận

Trong bối cảnh tình trạng ngập lụt đô thị tại thành phố Đà Nẵng ngày càng diễn biến phức tạp, đề tài đã bước đầu xây dựng thành công một hệ thống dự báo và cảnh báo ngập lụt ứng dụng công nghệ Edge AI. Mô hình học sâu MTS-LSTM được huấn luyện từ dữ liệu mưa, mực nước và khí tượng – thủy văn thực tế tại Đà Nẵng đã cho kết quả khả quan, với hệ số Nash-Sutcliffe Efficiency (NSE) đạt 0,73 và hệ số tương quan $r = 0,86$. Mô hình thể hiện tốt khả năng nhận diện các đỉnh lũ và có độ trễ thấp, phù hợp với yêu cầu cảnh báo sớm trong mùa mưa.

Việc tích hợp công nghệ Edge AI đã mang lại nhiều lợi ích thiết thực cho hệ thống, bao gồm: giảm đáng kể độ trễ xử lý xuống khoảng 100 mili-giây, tiết kiệm tới 97% băng thông so với phương pháp truyền thống, đồng thời tăng cường bảo mật thông tin nhờ khả năng xử lý dữ liệu cục bộ. Kiến trúc phân tán Edge-Cloud, với việc mô phỏng sử dụng thiết bị ESP32, đã chứng minh tính linh hoạt cao trong vận hành. Hệ thống cũng sẵn sàng tích hợp với các nền tảng đô thị thông minh như Danang Smart City trong tương lai.

Tuy nhiên, mô hình vẫn còn tồn tại một số hạn chế, đặc biệt là xu hướng làm trơn dữ liệu khiến đỉnh lũ bị dự báo thấp hơn thực tế, cũng như sự thiếu hụt dữ liệu nền trong mùa khô. Ngoài ra, tập dữ liệu huấn luyện còn hạn chế về thời gian (chỉ gồm hai năm), ảnh hưởng đến khả năng tổng quát hóa và làm tăng nguy cơ quá khớp.

Hướng phát triển

Trong tương lai, để nâng cao hiệu quả của hệ thống, nhóm đề xuất một số định hướng sau:

- **Mở rộng dữ liệu huấn luyện:** Bổ sung dữ liệu thủy văn nền, bao gồm dòng chảy cơ bản, thông tin thủy triều, và dữ liệu vận hành hồ chứa, nhằm cải thiện độ chính xác trong cả mùa mưa và mùa khô.
- **Mở rộng tập dữ liệu huấn luyện:** tăng thời gian của tập dữ liệu huấn luyện lên ít nhất 5 năm để tăng khả năng tổng quát hóa của mô hình MTS-LSTM, giảm nguy cơ overfitting, và dự báo chính xác hơn cho các sự kiện lũ bất thường.

- **Cải thiện mô hình AI:** Áp dụng các kỹ thuật hiệu chỉnh sau mô hình (post-processing) hoặc điều chỉnh hàm mất mát để tăng cường khả năng dự báo đỉnh lũ, đáp ứng tốt hơn yêu cầu cảnh báo lũ lớn.
- **Triển khai thực tế tại địa phương:** Xây dựng hệ thống thử nghiệm thực tế tại các khu vực ngập lụt nghiêm trọng như Thanh Khê, Cẩm Lệ, và Hải Châu nhằm kiểm nghiệm hiệu quả vận hành, độ tin cậy và khả năng tích hợp vào hạ tầng đô thị.
- **Tối ưu kiến trúc Edge AI:** Triển khai mô hình nhẹ hơn như TinyML hoặc TensorFlow Lite trên các thiết bị vi điều khiển như ESP32, giúp tiết kiệm điện năng và chi phí, đồng thời vẫn duy trì khả năng xử lý cục bộ hiệu quả.
- **Mở rộng tích hợp hệ sinh thái dữ liệu:** Tăng cường hợp tác với các đơn vị như Sở Tài nguyên và Môi trường, WATEC, và các viện nghiên cứu để tích hợp thêm dữ liệu và tối ưu hóa hệ thống.

Tóm lại, đề tài đã đặt nền móng cho một giải pháp dự báo lũ lụt đô thị tiên tiến, tận dụng công nghệ Edge AI và học sâu để hỗ trợ quản lý thiên tai tại Đà Nẵng. Với những cải tiến trong tương lai, hệ thống này hứa hẹn sẽ trở thành một công cụ hiệu quả, góp phần xây dựng một đô thị thông minh, an toàn, và bền vững hơn

TÀI LIỆU THAM KHẢO

[1]	D. King and C. Macgregor, “Using Social Indicators to Measure Community Vulnerability to Natural Hazards,” <i>Aust. J. Emerg. Manag.</i> , vol. 15, no. 3, pp. 52–57, 2000, https://www.researchgate.net/publication/279706088_Using_social_indicators_to_measure_community_vulnerability_to_natural_hazards
[2]	A. Hebb, “Floods: Mapping Vulnerability in the Upper Thames Watershed Under a Changing Climate”, Report XI, CFCAS Project of Water Resources Risk and Vulnerability to Changing Climatic Conditions, University of Western Ontario, London, 2007, cfcas-climate/reports/vulnerability_mapping_report
[3]	S. N. Jonkman, “Global Perspectives on Loss of Human Life Caused by Floods” <i>Nat Hazards</i> , vol. 34, no. 2, pp. 151–175, Feb. 2005, https://sci-hub.se/https://link.springer.com/article/10.1007/s11069-004-8891-3
[4]	NVIDIA Developer Blog, “AI for Climate, Energy, and Ecosystem Resilience at NVIDIA GTC 2025.”, developer.nvidia.com/blog/ai-for-climate-energy-and-ecosystem-resilience-at-nvidia-gtc-2025/
[5]	Lương Tuấn Anh và cộng sự (2015), “Bản đồ ngập lụt và bản đồ đánh giá mức độ dễ bị tổn thương do lũ trên các lưu vực sông Lam, Bến Hải - Thạch Hãn và Thu Bồn”, (2015) vjol.info.vn/index.php/JSTD/article/
[6]	Trần Duy Kiều, “Nghiên cứu quản lý lũ lớn lưu vực sông Lam”, Luận án tiến sĩ chuyên ngành Phát triển nguồn nước, (2012), https://vn-document.net/document/luan-an-tien-si-ky-thuat-nghien-cuu-quan-ly-lu-lon-luu-vuc-song-lam/9647745879
[7]	L. Alfieri et al., “A global network for operational flood risk reduction” <i>Environmental Science & Policy</i> , vol. 84, pp. 149–158, Jun. 2018, doi:10.1016/j.envsci.2018.03.014
[8]	F. Ghobadi and D. Kang, “Application of Machine Learning in Water Resources Management: A Systematic Literature Review” <i>Water</i> , vol. 15, no. 4, Art. no. 4, Jan. 2023, doi:10.3390/w15040620
[9]	F. Krim, M.A. Armin, “A Review of Hydrodynamic and Machine Learning Approaches for Flood Inundation Modeling” <i>Water</i> , vol. 15, no. 3, Art. no. 3, Jan. 2023, doi:10.3390/w15030566 .
[10]	M. Zounemat-Kermani, O. Batelaan, M. Fadaee, and R. Hinkelmann, “Ensemble machine learning paradigms in hydrology: A review” <i>Journal of Hydrology</i> , vol. 598, p. 126266, Jul. 2021, doi:10.1016/j.jhydrol.2021.126266
[11]	Hydrology and Earth System Sciences, “Deep learning methods for flood mapping: a review of existing applications and future research directions”, vol. 26, no. 16, pp. 4345–4378, Aug. 2022, https://hess.copernicus.org/articles/26/4345/2022/hess-26-4345-2022.pdf

[12]	W. J. Wee, N. B. Zaini, A. N. Ahmed, and A. El-Shafie, “A review of models for water level forecasting based on machine learning,” <i>Earth Sci Inform</i> , vol. 14, no. 4, pp. 1707–1728, Dec. 2021, doi:10.1007/s12145-021-00664-9
[13]	K. S. M. H. Ibrahim, Y. F. Huang, A. N. Ahmed, C. H. Koo, and A. El-Shafie, “A review of the hybrid artificial intelligence and optimization modelling of hydrological streamflow forecasting” <i>Alexandria Engineering Journal</i> , vol. 61, no. 1, pp. 279–303, Jan. 2022, doi:10.1016/j.aej.2021.04.100 .
[14]	H. Mosaffa, M. Sadeghi, I. Mallakpour, M. Naghdizadegan Jahromi, and H. R. Pourghasemi, “Chapter 43 - Application of machine learning algorithms in hydrology,” in <i>Computers in Earth and Environmental Sciences</i> , H. R. Pourghasemi, Ed., Elsevier, 2022, pp. 585–591. doi:10.1016/B978-0-323-89861-4.00027-0 .
[15]	Y. Guo, Y. Zhang, L. Zhang, and Z. Wang, “Regionalization of hydrological modeling for predicting streamflow in ungauged catchments: A comprehensive review,” <i>WIREs Water</i> , vol. 8, no. 1, p. e1487, 2021, doi:10.1002/wat2.1487 .
[16]	S. R. Sannasi Chakravarthy, N. Bharanidharan, and H. Rajaguru, “A systematic review on machine learning algorithms used for forecasting lake-water level fluctuations” <i>Concurrency and Computation: Practice and Experience</i> , vol. 34, no. 24, p. e7231, 2022, https://www.researchgate.net/publication/A_systematic_review_on_machine_learning_algorithms_used_for_forecasting_lake-water_level_fluctuations
[17]	Deltares, “Delft-FEWS Platform”, deltares.nl/en/software-and-data/products/delft-fews-platform?
[18]	G. Nearing, D. Cohen, V. Dube,...” <i>AI Increases Global Access to Reliable Flood Forecasts</i> ”, 2021, researchgate.net/publication/372785529_AI_Increases_Global_Access_to_Reliable_Flood_Forecasts
[19]	O.Ejike, D.Ndzi, “Climate change is making rain difficult to predict – we trained an AI to help tropical countries adapt”, 20/9/2022, climate-change-is-making-rain-difficult-to-predict-we-trained-an-ai-to-help-tropical-countries-adapt
[20]	A. N. Ahmed et al., “Water level prediction using various machine learning algorithms: a case study of Durian Tunggal river, Malaysia” <i>Engineering Applications of Computational Fluid Mechanics</i> , vol. 16, 10.1080/19942060.2021.2019128 .
[21]	Quang Đ. N., Chiêu T. Q., Huệ Đ. T., and Ngân N. T. K., “Prediction of Water Level in Kien Giang river using Regression-Based Model” 1, no. 80, Art. no. 80, Nov. 2022, DLU123456789/267060
[22]	T. Kusudo, A. Yamamoto, M. Kimura, and Y. Matsuno, “Development and Assessment of Water-Level Prediction Models for Small Reservoirs Using a

	<i>Deep Learning Algorithm,</i> ” Water, vol. 14, no. 1, Art. no. 1, Jan. 2022, doi:10.3390/w14010055.
[23]	W.-D. Guo, W.-B. Chen, S.-H. Yeh, C.-H. Chang, and H. Chen, “ <i>Prediction of River Stage Using Multistep-Ahead Machine Learning Techniques for a Tidal River of Taiwan,</i> ” Water, vol. 13, no. 7, Art. no. 7, Jan. 2021, doi:10.3390/w13070920.
[24]	SGGP Online, “ <i>Cảnh báo lũ đặc biệt lớn tại Quảng Nam và Đà Nẵng</i> ”, Oct. 10, 2022, sggp.org.vn/canh-bao-lu-dac-biet-lon-tai-quang-nam-va-da-nang-post573328.html
[25]	DaNang Online, “ <i>Lũ trên sông Yên, Cẩm Lệ đang dâng lên, cảnh báo mưa, lũ lớn</i> ”, 13/10/2022, https://baodanang.vn/channel/5403/lu-tren-song-yen-cam-le-dang-dang-len-canh-bao-mua-lu-lon-3926217
[26]	I. R. Widiyari, R. Efendi, “ <i>Utilizing LSTM-GRU for IOT-Based Water Level Prediction Using Multi-Variable Rainfall Time Series Data</i> ”, 24/7/2024 https://www.mdpi.com/2227-9709/11/4/73
[27]	N. Dong, “ <i>Da Nang searching for urban flooding solutions</i> ”, VnExpress International, https://e.vnexpress.net/news/news/environment/da-nang-searching-for-urban-flooding-solutions-4653915.html
[28]	Climates to travel, “ <i>Da Nang climate: weather by month, temperature, rain</i> ”, https://www.climatestotravel.com/climate/vietnam/da-nang
[29]	EBRD, “ <i>Flood Simulation Modelling: Da Nang, Vietnam</i> ”, https://www.ebrdgreencities.com/policy-tool/flood-simulation-modelling-da-nang-vietnam/
[30]	LSTM input dataset for each rainfall scenario generated through R-R... Download Scientific Diagram, https://www.researchgate.net/figure/LSTM-input-dataset-for-each-rainfall-scenario-generated-through-R-R-simulation_fig5_369138659
[31]	F. Krim, M. A. Armin, “ <i>A Review of Hydrodynamic and Machine Learning Approaches for Flood Inundation Modeling,</i> ” Water, vol. 15, no. 3, Art. no. 3, Jan. 2023, doi: 10.3390/w15030566.
[32]	A. Pranolo, “ <i>Enhanced Multivariate Time Series Analysis Using LSTM: A Comparative Study of Min-Max and Z Score Normalization Techniques</i> ”, August, 2024, https://www.researchgate.net/publication/387550499_Enhanced_Multivariate_Time_Series_Analysis_Using_LSTM_A_Comparative_Study_of_Min-Max_and_Z-Score_Normalization_Techniquesv
[33]	NeuralHydrology 1.12.0 documentation, “ <i>Multi-Timescale Prediction</i> ”, https://neuralhydrology.readthedocs.io/en/latest/tutorials/multi-timescale.html
[34]	HESS, “ <i>Rainfall–runoff prediction at multiple timescales with a single Long Short-Term Memory network</i> ”, 15 Oct 2020, https://hess.copernicus.org/articles/25/2045/2021/

[35-40, 42,44, 46,48]	Neuralhydrology, evaluation, metrics.py, https://github.com/neuralhydrology/neuralhydrology/blob/master/neuralhydrology/evaluation/metrics.py
[41, 43, 45, 47]	Neuralhydrology API, evaluation, metrics, https://neuralhydrology.readthedocs.io/en/latest/api/neuralhydrology/evaluation/metrics.html
[49]	P. Li, X. Wang, “Multi-Model Running Latency Optimization in an Edge Computing Paradigm”, 2022, www.mdpi.com/1424-8220/22/16
[50]	C. Yang, J. You, “Network Bandwidth Allocation Problem For Cloud Computing”, March 15 2022 , https://arxiv.org/pdf/2203.06725
[51]	K. Nanath, R. Pillal, “A Model for Cost-Benefit Analysis of Cloud Computing”, 11/08/2013, https://scholarworks.lib.csusb.edu/cgi/
[52]	S. Parikh, D. Dave, “Security and Privacy Issues in Cloud, Fog and Edge Computing”, 2019, https://www.sciencedirect.com/science/article/pii/