

# So sánh kiến trúc CNN và SNN tích hợp CAM trong phân tích tín hiệu điện tâm đồ

Nguyễn Thế Khôi<sup>a</sup>, Nguyễn Thị Minh Ly<sup>a</sup>, Đặng Minh Nguyệt<sup>a</sup>

<sup>a</sup>Sinh viên Viện Trí tuệ nhân tạo, VNU - UET

## Abstract

Phân tích tín hiệu điện tâm đồ (ECG) đóng vai trò quan trọng trong chẩn đoán sớm các bệnh lý tim mạch, đặc biệt trong bối cảnh các hệ thống hỗ trợ y tế thông minh ngày càng phát triển. Tuy nhiên, việc triển khai các mô hình học sâu như mạng nơ-ron tích chập (CNN) trên các thiết bị biên gặp nhiều thách thức do hạn chế về tài nguyên và năng lượng. Nghiên cứu này hướng đến việc đánh giá khả năng thay thế của mạng nơ-ron xung (SNN) nhằm giảm thiểu tiêu thụ năng lượng trong phân loại tín hiệu ECG mà vẫn giữ được hiệu quả phân tích. Đồng thời, cơ chế chú ý theo kênh (Channel Attention Mechanism – CAM) được tích hợp nhằm tăng cường khả năng học đặc trưng của mô hình bằng cách làm nổi bật các kênh thông tin quan trọng trong tín hiệu. Dữ liệu được tiền xử lý từ bộ PTB-XL, đảm bảo tính nhất quán và phù hợp cho huấn luyện. Bài báo tập trung vào so sánh hai kiến trúc CNN và SNN+CAM trên cùng dữ liệu đầu vào, làm rõ tiềm năng ứng dụng của SNN trong các hệ thống ECG tiết kiệm năng lượng.

**Keywords:** ECG, Mạng nơ-ron tích chập, Mạng nơ-ron xung, CAM, Tiết kiệm năng lượng; Phân loại tín hiệu sinh học.

## 1. Giới thiệu

Tín hiệu điện tâm đồ (ECG) là một công cụ chẩn đoán không xâm lấn quan trọng, ghi lại hoạt động điện của tim thông qua các điện cực gắn trên da. Các đặc trưng như sóng P, phức hợp QRS và sóng T phản ánh trạng thái sinh lý của tim và hỗ trợ phát hiện bất thường như nhồi máu cơ tim, rối loạn nhịp và phì đại thất [1]. Nhờ chi phí thấp, dễ triển khai và giá trị lâm sàng cao, ECG được sử dụng rộng rãi trong giám sát và chẩn đoán tim mạch.

Gần đây, các phương pháp học sâu, đặc biệt là mạng nơ-ron tích chập (CNN), đã đạt nhiều tiến bộ trong phân tích tự động tín hiệu ECG, cho phép học đặc trưng trực tiếp từ tín hiệu thô và đạt độ chính xác cao [2, 3]. Tuy nhiên, chi phí tính toán và tiêu thụ năng lượng cao là rào cản lớn khi triển khai trên thiết bị biên hoặc hệ thống nhúng.

Mạng nơ-ron xung (Spiking Neural Networks – SNN) được đề xuất như một giải pháp thay thế sinh học và tiết kiệm năng lượng hơn, mô phỏng cơ chế xử lý theo xung của hệ thần kinh [4, 5]. Nhờ xử lý sự kiện rời rạc theo thời gian, SNN phù hợp với các ứng dụng thời gian thực trong y tế di động.

Bên cạnh hiệu năng, khả năng giải thích là yếu tố quan trọng trong môi trường lâm sàng. Thay vì các kỹ thuật trực quan hóa như Grad-CAM, nghiên cứu này tích hợp

cơ chế chú ý theo kênh (Channel Attention Mechanism – CAM) để làm nổi bật các kênh thông tin quan trọng trong tín hiệu và tăng cường khả năng học đặc trưng.

Nghiên cứu tiến hành so sánh giữa mô hình CNN truyền thống và SNN tích hợp CAM trên dữ liệu PTB-XL, đánh giá qua độ chính xác và hiệu quả năng lượng, nhằm làm rõ tiềm năng ứng dụng trong các hệ thống y tế thông minh.

## 2. Định hướng và Đóng góp chính

### 2.1. Công trình liên quan

Nhiều nghiên cứu đã tập trung khai thác tín hiệu điện tâm đồ dựa trên hai hướng tiếp cận chính: các kỹ thuật xử lý tín hiệu truyền thống và các mô hình học máy hiện đại. Các phương pháp truyền thống như biến đổi wavelet, biến đổi Fourier nhanh (FFT) và các bộ lọc số được sử dụng để trích xuất thủ công các đặc trưng tín hiệu [6]. Tuy nhiên, các kỹ thuật này phụ thuộc nhiều vào kinh nghiệm chuyên môn và khả năng khái quát hóa còn hạn chế trong các tình huống phức tạp.

Cũng như đã được đề cập ở phần trước, phần lớn các mô hình học sâu cho độ chính xác khá cao, tuy nhiên lại không phù hợp để triển khai ở những môi trường hạn chế về năng lượng và bộ nhớ. Một số hướng tiếp cận đã được đề xuất để giảm chi phí tính toán, chẳng hạn như rút gọn mô hình, nén trọng số, hoặc offloading lên đám mây [7].

Tuy vậy, những phương pháp này gặp khó khăn trong điều kiện mạng không ổn định hoặc khi lượng dữ liệu ECG cần truyền quá lớn. Sự ra đời của SNN đã mở ra tiềm năng cho một phương pháp vừa có độ chính xác cao, vừa tiết kiệm chi phí tính toán. Dù vậy, hầu hết các nghiên cứu ứng dụng SNN cho phân loại ECG vẫn còn hạn chế ở quy mô dữ liệu nhỏ hoặc chỉ sử dụng lại trọng số từ các mô hình CNN được huấn luyện trước, thay vì huấn luyện trực tiếp với gradient thay thế — một hướng tiếp cận quan trọng để tận dụng tối đa tiềm năng học biểu diễn của SNN.

## 2.2. Đóng góp chính

Trong nghiên cứu này, nhóm tập trung phân tích và so sánh hiệu quả giữa hai kiến trúc điển hình: CNN – đại diện cho mạng nơ-ron truyền thống thế hệ thứ hai, và SNN – đại diện cho mạng nơ-ron thế hệ thứ ba, lấy cảm hứng từ cách thức xử lý thông tin sinh học của não bộ. Cả hai mô hình được huấn luyện trực tiếp trên tập dữ liệu ECG PTB-XL, với quy trình tiền xử lý và đánh giá thống nhất nhằm đảm bảo tính khách quan khi so sánh.

Trong khi CNN sử dụng tín hiệu liên tục và khai thác khả năng học đặc trưng thông qua tích chập không gian, thì SNN truyền tải và xử lý thông tin bằng các xung nhị phân, mô phỏng gần đúng cơ chế phát xung và tích lũy điện thế của tế bào thần kinh sinh học. Nhờ đặc tính kích hoạt thưa và chỉ phát xung khi cần thiết, SNN sở hữu tiềm năng lớn trong việc giảm thiểu tiêu thụ năng lượng và tài nguyên tính toán [8]— yếu tố đặc biệt quan trọng trong các ứng dụng y sinh học triển khai trên thiết bị đeo hoặc hệ thống nhúng. Ngoài việc là hướng nghiên cứu quan trọng trong lĩnh vực khoa học thần kinh tính toán, SNN cũng đã chứng minh tính hiệu quả trong nhiều tác vụ xử lý tín hiệu sinh lý như ảnh y tế, điện não đồ (EEG) và điện dạ dày (EGG).

Thông qua việc đối chiếu hiệu năng, độ chính xác và khả năng triển khai trong điều kiện tài nguyên hạn chế, nghiên cứu nhằm đánh giá tiềm năng của SNN như một phương án thay thế CNN trong các hệ thống phân loại tín hiệu ECG thân thiện với năng lượng.

Tóm lược lại, những đóng góp chính của nghiên cứu bao gồm:

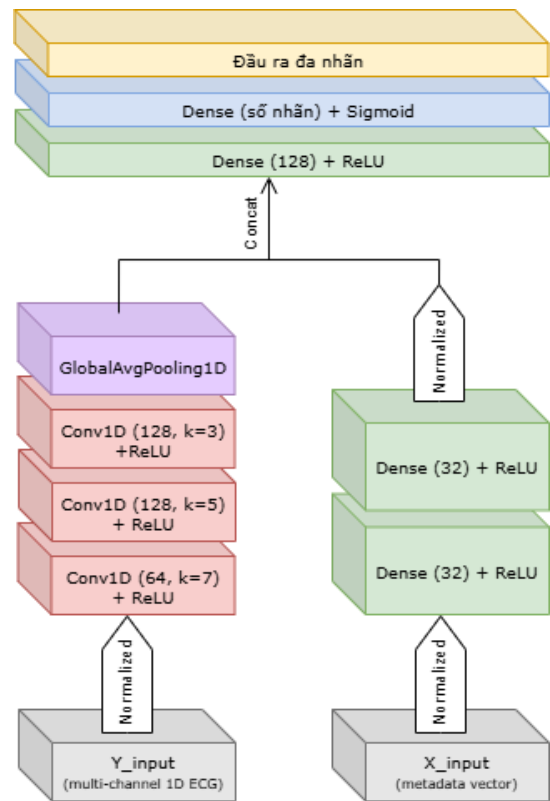
1. Đề xuất một kiến trúc kết hợp giữa SNN và CAM (cơ chế chú ý theo kênh), nhằm tăng cường khả năng tập trung vào các vùng tín hiệu quan trọng trong khi vẫn tối ưu hóa chi phí tính toán.
2. So sánh trực tiếp độ chính xác và gián tiếp hiệu quả tiêu thụ năng lượng giữa CNN và SNN trong tác vụ phân loại tín hiệu ECG thô, qua đó làm rõ ưu điểm và

hạn chế của từng mô hình trong bối cảnh triển khai thực tế.

3. Khai thác bộ dữ liệu ECG PTB-XL, một trong những tập dữ liệu ECG lớn và toàn diện nhất hiện nay, bao phủ đa dạng các loại tín hiệu và chẩn đoán. Nhờ bao gồm 12 chuyển đạo, tập dữ liệu này phù hợp để phát triển các mô hình cho cả ứng dụng lâm sàng và phi lâm sàng.
4. Ứng dụng huấn luyện trực tiếp SNN bằng gradient xấp xỉ (surrogate gradient), thay vì dựa vào phương pháp chuyển đổi mô hình ANN sang SNN như trong các hướng tiếp cận truyền thống, giúp khai thác trọn vẹn khả năng học sâu của mạng SNN.

## 3. Mô hình

### 3.1. Mạng nơ-ron tích chập



Hình 1: Kiến trúc mạng nơ-ron tích chập

Kiến trúc mạng CNN được thiết kế với mục tiêu trích xuất đặc trưng không gian–thời gian từ tín hiệu sinh lý và kết hợp với thông tin phi tín hiệu để thực hiện phân loại đa nhãn. Mô hình có cấu trúc hai nhánh tách biệt: một nhánh xử lý chuỗi tín hiệu và một nhánh xử lý dữ liệu metadata (Hình 1).

*Nhánh tín hiệu* gồm ba lớp tích chập một chiều (*Conv1D*) có số lượng bộ lọc lần lượt là 64, 128, và 128, với kích thước kernel tương ứng là 7, 5 và 3. Mỗi lớp tích chập được theo sau bởi hàm kích hoạt ReLU nhằm tăng tính phi tuyến cho mô hình. Sau các lớp này, một lớp gộp trung bình toàn cục (*GlobalAveragePooling1D*) được sử dụng để giảm chiều dữ liệu, giúp giảm số lượng tham số và tăng khả năng tổng quát.

*Nhánh phi tín hiệu* bao gồm hai lớp Dense liên tiếp, mỗi lớp có 32 đơn vị ẩn với hàm kích hoạt ReLU. Đầu vào là một vector đặc trưng từ dữ liệu nhân khẩu học và lâm sàng đã được chuẩn hóa.

Hai nhánh được kết nối thông qua một lớp ghép nối (*Concatenate*), tiếp theo là một lớp Dense có 128 đơn vị ẩn sử dụng ReLU. Lớp đầu ra gồm 5 đơn vị tương ứng với năm nhãn bệnh lý, sử dụng hàm kích hoạt sigmoid để dự đoán xác suất xuất hiện của mỗi nhãn.

Công thức tổng quát cho lớp tích chập được biểu diễn như sau:

$$y_i^{(k)} = \sum_{j=0}^{C-1} \sum_{l=0}^{L-1} w_l^{(k,j)} \cdot x_{i+l}^{(j)} + b^{(k)} \quad (1)$$

với  $x_{i+l}^{(j)}$  là đầu vào tại vị trí  $i + l$  của kênh  $j$ ,  $w_l^{(k,j)}$  là trọng số kernel, và  $y_i^{(k)}$  là đầu ra tại kênh  $k$ . Hàm kích hoạt ReLU được áp dụng sau mỗi tích chập:

$$\text{ReLU}(x) = \max(0, x) \quad (2)$$

Lớp đầu ra sử dụng sigmoid:

$$\hat{y}_k = \sigma(z_k) = \frac{1}{1 + e^{-z_k}} \quad (3)$$

Ưu điểm chính của kiến trúc này là khả năng học biểu diễn từ dữ liệu tín hiệu có cấu trúc chuỗi, đồng thời khai thác thông tin bổ sung từ các thuộc tính phi tín hiệu. Việc sử dụng cấu trúc đơn giản, không có các thành phần phụ thuộc trạng thái như RNN hoặc attention giúp giảm chi phí tính toán và thời gian huấn luyện.

Tuy nhiên, hạn chế của mô hình là thiếu cơ chế tập trung (attention) hoặc vùng nhận dạng rõ ràng (receptive field control) để diễn giải các đặc trưng học được. Ngoài ra, các lớp tích chập truyền thống có thể chưa tối ưu trong việc phát hiện các mẫu biến thiên phức tạp theo thời gian như trong tín hiệu ECG đa kênh. Việc mở rộng mô hình bằng các kỹ thuật như Channel Attention Mechanism (CAM) hoặc chuyển đổi sang SNN có thể giúp cải thiện tính giải thích và độ chính xác trong các ứng dụng lâm sàng.

### 3.2. Mạng nơ-ron xung và Cơ chế chú ý theo kênh

#### 3.2.1. Mạng nơ-ron xung

Trong các mô hình mạng nơ-ron tích chập có phát xung (Spiking Neural Networks – SNN), Leaky Integrate-and-Fire (LIF) là một trong những mô hình nơ-ron được sử dụng phổ biến nhất [9]. Hoạt động của nơ-ron LIF có thể hiểu đơn giản qua ba giai đoạn:

1. Integrate (tích lũy): nơ-ron liên tục nhận và cộng dồn các tín hiệu đầu vào theo thời gian, làm thay đổi điện thế màng.
2. Leaky (rò rỉ): điện thế màng không được giữ nguyên, mà sẽ giảm dần theo thời gian do hiệu ứng “rò” giống dòng điện qua điện trở.
3. Fire (phát xung): nếu điện thế vượt qua một ngưỡng định trước, nơ-ron sẽ phát ra một xung nhị phân (spike), sau đó điện thế màng sẽ được đưa về lại mức khởi đầu (reset).

Mô hình LIF truyền tải thông tin bằng các xung rời rạc và không khả vi, điều này khiến cho phương pháp lan truyền ngược truyền thống (backpropagation) không thể áp dụng trực tiếp [9].

Để giải quyết bài toán này, Lee và các đồng sự đề xuất thuật toán lan truyền ngược sử dụng quy tắc cập nhật bội (multiplicative update rules) [10]. Panda và các đồng sự sau đó phát triển một mô hình autoencoder spiking tích chập dạng phân cấp và huấn luyện nó bằng phương pháp backpropagation [11]. Một hướng đi khác phổ biến là huấn luyện một mô hình CNN thông thường rồi chuyển đổi thành SNN, từ đó tránh được việc huấn luyện trực tiếp SNN [12].

Tuy nhiên, trong nghiên cứu này, nhóm lựa chọn huấn luyện trực tiếp mạng SNN, bằng cách áp dụng surrogate gradient — một kỹ thuật sử dụng các hàm khả vi gần đúng (ví dụ: `sigmoid`, `atan`, `fast sigmoid`...) để thay thế cho đạo hàm không xác định của hàm kích hoạt phát xung, cho phép thực hiện lan truyền ngược qua các tầng SNN.

Mô hình nơ-ron LIF đơn giản bao gồm ba thành phần chính: nơ-ron tiền khớp thần kinh  $i$ , khớp thần kinh (nối giữa hai nơ-ron) và nơ-ron hậu khớp thần kinh  $j$ .

Về mặt sinh học, hành vi của một nơ-ron LIF được mô tả bằng phương trình vi phân, trong đó điện thế màng tích lũy dòng điện rò và các xung đầu vào từ các nơ-ron  $i$  trước khớp thần kinh. Phương trình vi phân được biểu diễn như sau:

$$\tau \frac{dU_j(t)}{dt} = -U_j(t) + RI(t) \quad (4)$$

với  $U_j(t)$  là điện thế màng của nơ-ron hậu khớp thần kinh  $j$  tại thời điểm  $t$ ;  $\tau$  là hằng số thời gian màng, thường được

định nghĩa là  $\tau = RC$ , trong đó  $R$  là điện trở màng và  $C$  là điện dung màng;  $I(t)$  là dòng điện đầu vào tại thời điểm  $t$ .

Tuy nhiên, vì mô phỏng trên máy tính là rời rạc, phương trình vi phân cần được chuyển đổi thành phương trình sai phân để tính toán: Cụ thể, tại mỗi thời điểm  $t$ , điện thế màng của nơ-ron hậu khớp thần kinh  $j$  được cập nhật theo công thức:

$$U_j(t) = \lambda U_j(t-1) + \sum_i w_{ij} \cdot U_i(t) \quad (5)$$

Trong đó,  $U_j(t)$  là điện thế màng của nơ-ron hậu khớp thần kinh  $j$  tại thời điểm  $t$ ;  $\lambda U_j(t-1)$  là phần điện thế còn lại từ thời điểm trước đó  $t-1$ , sau khi đã bị rò rỉ một phần (do  $0 < \lambda < 1$ );  $U_i(t)$  là đầu ra từ nơ-ron tiền khớp thần kinh  $i$  tại thời điểm  $t$ ;  $w_{ij}$  là trọng số từ nơ-ron  $i$  đến nơ-ron  $j$ .

Do đầu ra  $U_i(t)$  chỉ là nhị phân, phép nhân  $w_{ij}U_i(t)$  trở thành phép cộng có điều kiện: nếu  $U_i(t) = 1$  thì cộng  $w_{ij}$ , còn nếu  $U_i(t) = 0$  thì bỏ qua.

Khi điện thế  $U_j(t)$  vượt qua một ngưỡng định sẵn  $v_{th}$ , nơ-ron phát ra xung có công thức như sau:

$$\text{output}(t) = \begin{cases} 1, & \text{if } U_j(t) > v_{th} \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

Ngay sau khi xung được phát, điện thế  $U_j(t)$  sẽ được đưa về mức reset  $v_r$  (thường chọn là 0), và nơ-ron bước vào giai đoạn trơ (refractory period) — tức không phản hồi với tín hiệu mới trong một khoảng thời gian ngắn, trước khi quay lại chu trình xử lý tiếp theo.

### 3.2.2. Cơ chế chú ý theo kênh

Cơ chế chú ý theo kênh (Channel Attention Mechanism) là một cơ chế chú ý nhằm học trọng số cho các kênh đặc trưng, giúp xác định mức độ quan trọng của từng kênh trong quá trình phân loại.

Trong bài toán phân loại bất thường tìm sử dụng tín hiệu ECG, nhóm áp dụng hai phép pooling toàn cục: *GlobalAveragePooling* (phép pooling trung bình), và *GlobalMaxPool* (phép pooling cực đại) dọc theo trục thời gian. Các phép này giúp nén thông tin của mỗi kênh thành một giá trị duy nhất. Sau đó, hàm sigmoid được sử dụng để học mặt nạ trọng số cho từng kênh, qua đó điều chỉnh mức độ đóng góp của mỗi kênh vào quá trình phân loại.

Sau lớp SNN thứ hai, tín hiệu ECG đầu vào được chuyển thành các đặc trưng cục bộ dưới dạng tensor  $F \in \mathbb{R}^{C \times W}$ , trong đó  $C = 128$  là số lượng kênh đặc trưng sau hai lớp tích chập và  $W$  là chiều dài tín hiệu sau các phép pooling.

Mặc dù đặc trưng này đã được tổng hợp từ toàn bộ các kênh, mối quan hệ giữa các kênh vẫn chưa được khai thác rõ ràng do tính chất cục bộ của phép tích chập. Vì vậy, nhóm lựa chọn chèn module chú ý theo kênh ngay sau lớp SNN thứ hai để tăng cường khả năng học trọng số cho từng kênh.

Việc tích hợp cơ chế chú ý theo kênh ngay sau hai lớp SNN đầu tiên trong mô hình không phải ngẫu nhiên mà được quyết định dựa trên đặc điểm của quá trình trích xuất đặc trưng và mối quan hệ giữa các kênh tín hiệu [13]. Nếu CAM được chèn quá sớm — ví dụ sau lớp SNN đầu tiên hoặc ngay trước các lớp tích chập — thì tín hiệu đầu vào vẫn ở dạng thô, chưa qua xử lý phi tuyến, và chưa hình thành được biểu diễn không gian có ý nghĩa. Điều này có thể khiến trọng số chú ý trở nên nhiễu hoặc thiếu chính xác, do không đủ thông tin để xác định mức độ quan trọng giữa các kênh.

Ngược lại, khi tín hiệu đầu vào trải qua hai lớp tích chập và hai tầng nơ-ron LIF phi tuyến, các đặc trưng thu được đã được xử lý sâu, phản ánh rõ ràng hơn các mẫu hình cục bộ cũng như mối quan hệ không gian giữa các kênh. Vào thời điểm này, tín hiệu chứa nhiều thông tin hơn về cấu trúc dữ liệu, giúp CAM học được trọng số chú ý hiệu quả, làm nổi bật các kênh có đóng góp quan trọng hơn.

### 3.2.3. Mô hình SNN tích hợp CAM

Nhóm đề xuất mô hình học sâu lai kết hợp giữa mạng nơ-ron xung (Spiking Neural Network - SNN) và cơ chế chú ý theo kênh (Channel Attention Mechanism - CAM), được thiết kế đặc biệt cho bài toán phân loại tín hiệu điện tâm đồ (ECG) (Hình 2). Mô hình này xử lý tín hiệu đầu vào một chiều (1-D ECG) và sử dụng các lớp SNN thay cho các lớp kích hoạt truyền thống, giúp tiết kiệm năng lượng. Đồng thời, CAM được tích hợp để nâng cao khả năng trích xuất đặc trưng và cải thiện hiệu suất phân loại. Cấu trúc mô hình bao gồm:

#### 1. Khối tích chập xung (Spiking Convolutional Block):

Mô hình sử dụng ba lớp tích chập 1-D với số lượng kênh tăng dần từ 64, 128 đến 256, mỗi lớp được theo sau bởi một lớp LIFNode — nơ-ron xung sử dụng hàm thay thế sigmoid để lan truyền gradient. Các lớp tích chập này sử dụng kernel size = 3 và padding = 1, giúp giữ nguyên chiều dài tín hiệu sau mỗi lớp. Sau lớp thứ hai, CAM được chèn vào để tính toán trọng số cho các kênh, ưu tiên các kênh chứa thông tin quan trọng trước khi đi qua lớp tích chập thứ ba. Mặc dù kiến trúc sử dụng các lớp Conv1d thường thấy trong mạng CNN, các lớp này đều được tích hợp với nơ-ron xung kiểu LIFNode cùng cơ chế lan truyền gradient



Bảng 1: Thông tin các siêu lớp trong bộ dữ liệu PTB-XL

Số lượng bản	Siêu lớp	Mô tả
9528	NORM	Bình thường
5486	MI	Nhồi máu cơ tim
5250	STTC	Thay đổi ST/T
4907	CD	Rối loạn dẫn truyền
2655	HYP	Tăng huyết áp

số nhân sẽ vượt quá số lượng bản ghi do một bản ghi có thể chứa nhiều chẩn đoán đồng thời.

Dữ liệu sóng ECG được lưu trữ theo định dạng WaveForm DataBase (WFDB) với độ phân giải 16 bit tại độ nhạy  $1\mu\text{V}/\text{LSB}$  và tần số lấy mẫu 500Hz. Ngoài ra, nhóm phát triển cũng cung cấp phiên bản dữ liệu đã giảm tần số xuống 100Hz. Bộ dữ liệu còn được bổ sung các thông tin siêu dữ liệu phong phú, bao gồm dữ liệu nhân khẩu học, đặc điểm nhồi máu cơ tim, xác suất xảy ra các chẩn đoán ECG, cũng như các đặc trưng tín hiệu đã được chú thích.

#### 4.2. Tiền xử lý dữ liệu

Trong nghiên cứu này, do giới hạn về năng lực và tài nguyên tính toán, nhóm lựa chọn sử dụng các bản ghi ECG với tần số lấy mẫu là 100Hz. Dữ liệu sóng điện tâm đồ được trích xuất bằng thư viện `wfdb` thông qua hàm `rdsamp()`, trả về một tuple gồm tín hiệu (`signal`) và siêu dữ liệu (`meta`). Trong đó, `signal` là một mảng NumPy hai chiều biểu diễn biên độ tín hiệu theo thời gian trên 12 chuyển đạo. Các tín hiệu này sau đó được gom lại thành một mảng ba chiều có kích thước tương ứng với số lượng mẫu, độ dài mỗi tín hiệu (số bước thời gian), và số kênh (chuyển đạo). Kết quả thu được là một tensor có shape (21837, 1000, 12) — tương ứng với 21.837 bản ghi ECG, mỗi bản dài 10 giây (1000 điểm tại tần số lấy mẫu 100Hz) và gồm 12 chuyển đạo.

Tiếp theo, nhóm tải siêu dữ liệu từ tệp `ptb_xl_database.csv`, trong đó các nhãn chẩn đoán `scp_codes` được phân tích thành định dạng danh sách nhãn đa lớp. Đồng thời, nhóm cũng sử dụng tệp `scp_statements.csv` để ánh xạ ý nghĩa của từng mã SCP và lọc ra các nhãn mang tính chẩn đoán (có `diagnostic == 1`).

Thông qua việc phân tích thống kê, nhóm nhận thấy rằng 74.5% các bản ghi chỉ có một nhãn chẩn đoán chính (superclass), trong khi khoảng 25.5% còn lại có từ hai đến bốn nhãn, cho thấy mức độ đa nhãn khá cao trong tập dữ liệu. Tương tự, đối với subclass, hơn 30% các bản ghi mang từ hai subclass trở lên. Sau đó, nhóm tiến hành mã

hóa one-hot các nhãn để phục vụ quá trình huấn luyện và đánh giá mô hình, đồng thời thực hiện các phân tích bổ sung về phân phối nhãn theo các thuộc tính nhân khẩu học và kỹ thuật như tuổi, chiều cao, cân nặng, giới tính, cơ sở y tế, loại thiết bị đo, v.v.

Theo khuyến nghị từ nguồn dữ liệu gốc, tập dữ liệu PTB-XL đã được chia thành 10 fold phân tầng, trong đó các bản ghi thuộc cùng một bệnh nhân được đảm bảo nằm trong cùng một fold. Các fold 9 và 10 được kiểm định thủ công và có chất lượng nhãn cao hơn, vì vậy thường được sử dụng cho validation và test. Dựa vào đó, nhóm quyết định sử dụng fold 1–8 cho huấn luyện, fold 9 cho kiểm định, và fold 10 cho đánh giá cuối cùng.

#### 4.3. Cài đặt và Huấn luyện

Toàn bộ quy trình xây dựng, huấn luyện và đánh giá mô hình được thực hiện trên nền tảng PyTorch, một thư viện deep learning linh hoạt, cho phép định nghĩa kiến trúc mạng, quản lý dữ liệu và tối ưu hóa đa dạng.

Để đảm bảo tính khách quan và khả năng tái lập, nhóm giữ nguyên toàn bộ các siêu tham số trong cả hai quá trình huấn luyện. Cụ thể, nhóm sử dụng optimizer Adam với tốc độ học (learning rate) ban đầu  $1 \times 10^{-2}$ , với scheduler StepLR với hệ số giảm  $\gamma = 0.1$  sau mỗi `step_size=5` epochs. Số epoch huấn luyện là 20 và batch-size là 64.

Với bài toán phân loại đa nhãn (multi-label), đầu ra của mạng SNN là logits kích thước `[batch_size, num_labels]`. Nhóm sử dụng hàm mất mát BCEWithLogitsLoss, tính loss riêng cho từng nhãn rồi lấy trung bình. Điều này giúp mô hình học cân đối giữa các lớp ngay cả khi tần suất xuất hiện không đồng đều. Hàm mất mát có công thức như sau:

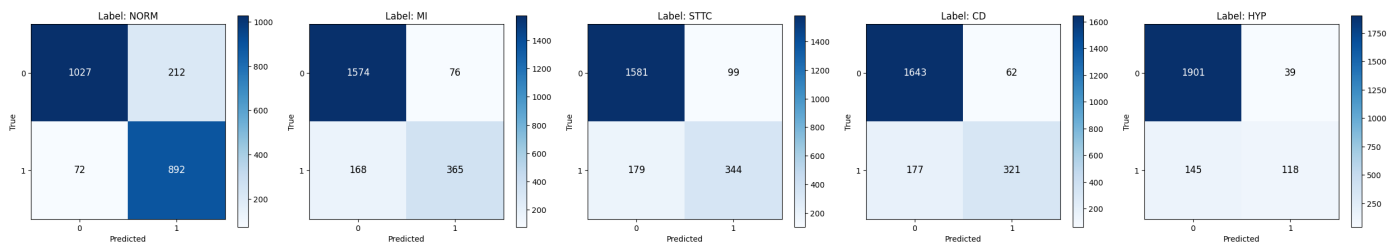
$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N \left[ \max(x_i, 0) - x_i y_i + \log(1 + e^{-|x_i|}) \right] \quad (7)$$

trong đó  $N$  là số lượng mẫu,  $y_i$  là nhãn thực của mẫu thứ  $i$ , và  $x_i$  là logit (đầu ra thô chưa qua hàm sigmoid) cho mẫu thứ  $i$ .

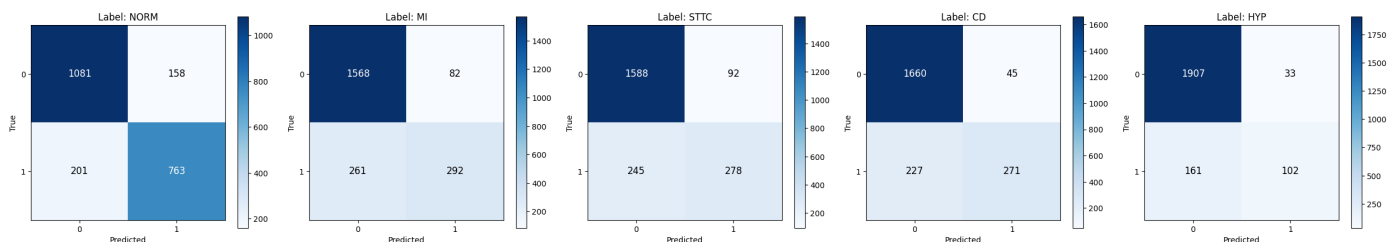
Với mô hình CNN, nhóm sử dụng hàm Binary Cross Entropy với công thức như sau:

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^N [y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)] \quad (8)$$

trong đó  $N$  là số lượng mẫu,  $y_i$  là nhãn thực, và  $\hat{y}_i$  là xác suất dự đoán tương ứng.



Hình 3: Ma trận nhầm lẫn của mô hình CNN



Hình 4: Ma trận nhầm lẫn của mô hình SNN tích hợp CAM

## 5. Kết quả và Thảo luận

### 5.1. Chỉ số đánh giá

Các chỉ số được sử dụng để đánh giá hiệu quả của mô hình phân loại bao gồm: *Accuracy*, *Precision*, *Recall* và *F1-score*. Cụ thể:

1. *Accuracy* (Độ chính xác): Là tỉ lệ giữa số mẫu được dự đoán đúng trên tổng số mẫu. *Accuracy* phản ánh hiệu suất tổng thể của mô hình trong việc phân loại đúng.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

2. *Precision* (Độ chính xác theo lớp dương): Là tỉ lệ giữa số mẫu dương được dự đoán đúng trên tổng số mẫu được dự đoán là dương. *Precision* phản ánh độ tin cậy của mô hình khi dự đoán một mẫu là dương.

$$Precision = \frac{TP}{TP + FP}$$

3. *Recall* (Độ bao phủ): Là tỉ lệ giữa số mẫu dương được dự đoán đúng trên tổng số mẫu thực sự là dương. *Recall* thể hiện khả năng phát hiện các trường hợp dương của mô hình.

$$Recall = \frac{TP}{TP + FN}$$

4. *F1-score*: Là trung bình điều hòa giữa *precision* và *recall*, giúp cân bằng hai yếu tố này. *F-score* đặc biệt

hữu ích trong trường hợp dữ liệu không cân bằng giữa các lớp.

$$F1\text{-score} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Trong đó, *TP* (*True Positive*) là số mẫu thực sự dương và được mô hình dự đoán đúng là dương, *FP* (*False Positive*) là số mẫu thực sự âm nhưng bị mô hình dự đoán nhầm là dương, *TN* (*True Negative*) là số mẫu thực sự âm và được mô hình dự đoán đúng là âm, *FN* (*False Negative*) là số mẫu thực sự dương nhưng bị mô hình dự đoán nhầm là âm.

### 5.2. Kết quả so sánh giữa hai mô hình đề xuất

Hình 3 và Hình 4 minh họa ma trận nhầm lẫn của hai mô hình: mạng nơ-ron tích chập (CNN) và mạng nơ-ron xung (SNN) có tích hợp cơ chế chú ý theo kênh (CAM) trong phân loại năm siêu lớp.

Dựa trên các ma trận này, Bảng 2 và Bảng 3 tổng hợp kết quả chi tiết tương ứng của hai mô hình.

Bảng 2: Kết quả mô hình CNN

Siêu lớp	Accuracy	Precision	Recall	F1-score
NORM	0.87	0.81	0.93	0.86
MI	0.89	0.84	0.70	0.76
STTC	0.87	0.78	0.66	0.71
CD	0.89	0.84	0.64	0.73
HYP	0.92	0.75	0.45	0.56

Bảng 3: Kết quả mô hình SNN tích hợp CAM

Siêu lớp	Accuracy	Precision	Recall	F1-score
NORM	0.84	0.83	0.79	0.81
MI	0.84	0.78	0.53	0.63
STTC	0.86	0.75	0.53	0.62
CD	0.89	0.86	0.54	0.67
HYP	0.92	0.76	0.39	0.51

Bảng 4 đưa ra so sánh giữa hai mô hình dựa trên giá trị trung bình theo macro của các chỉ số.

Bảng 4: Kết quả so sánh hai mô hình

Mô hình	Accuracy	Precision	Recall	F1-score
CNN	0.89	0.80	0.68	0.72
SNN+CAM	0.87	0.80	0.56	0.65

Kết quả cho thấy độ chính xác tổng thể (Accuracy) và độ chính xác theo lớp dương (Precision) của hai mô hình là tương đương, với sự khác biệt không đáng kể. Tuy nhiên, Recall trung bình của mô hình CNN vượt trội hơn khoảng 10% so với mô hình SNN tích hợp CAM, từ đó kéo theo F1-score của CNN cũng cao hơn. Điều này cho thấy CNN có khả năng nhận diện tốt hơn các trường hợp thuộc lớp dương.

### 5.3. Kết quả so sánh giữa mô hình SNN tích hợp CAM và một số mô hình khác

Mô hình SNN tích hợp cơ chế chú ý CAM được so sánh với ba mô hình khác dựa trên chỉ số AUROC (Bảng 5), trên cùng tập dữ liệu PTB-XL với các bản ghi có tần số lấy mẫu 100Hz [18]. Kết quả cho thấy SNN+CAM đạt  $AUROC = 0.91$ , không chênh lệch đáng kể so với các mô hình nổi bật như Inception1D [19] và LSTM [20], đồng thời vượt trội hơn mô hình Wavelet+NN [21]. Điều này cho thấy phương pháp SNN+CAM do nhóm đề xuất có hiệu quả cạnh tranh cao trong nhiệm vụ phân loại tín hiệu điện tim.

Bảng 5: Kết quả so sánh giữa mô hình SNN và một vài mô hình khác

Mô hình	AUROC
Inception 1D	0.92
LSTM	0.91
Wavelet+NN	0.87
SNN+CAM	0.91

### 5.4. Thảo luận

Một trong những thách thức chính mà nhóm gặp phải là chưa thể trực tiếp đo đặc mức tiêu thụ năng lượng của hai mô hình thông qua thực nghiệm. Dù vậy, dựa trên các nghiên cứu gần đây, nhóm vẫn kỳ vọng vào tiềm năng tiết kiệm năng lượng của SNN. Trong bài báo của mình, Xing và cộng sự đã tổng hợp những nghiên cứu sử dụng mô hình học sâu trong phân loại tín hiệu điện tâm đồ, và tìm ra năng lượng tiêu thụ trung bình của một số mô hình nổi bật. Cụ thể, mô hình CNN sử dụng FFT tiêu tốn 37mJ/nhịp, Wavelet kết hợp MLP là 4mJ/nhịp, còn phiên bản SNN chỉ tốn 1.78 $\mu$ J/nhịp [13]. Tương tự, nghiên cứu của Yan và cộng sự cho thấy, khi triển khai mô hình hai giai đoạn, mức tiêu thụ điện năng của SNN chỉ là 0.077W, trong khi CNN lên đến 10.40W [22].

Dù vậy, một số nghiên cứu khác cũng chỉ ra rằng hiệu quả năng lượng của SNN chỉ thực sự phát huy dưới những điều kiện tối ưu và nghiêm ngặt về kiến trúc và phần cứng [23]. Có thể thấy, các nghiên cứu về SNN vẫn còn mới mẻ và đang trong quá trình hoàn thiện, đòi hỏi thêm các thực nghiệm cụ thể để xác định rõ ràng lợi thế của mô hình này trong các ứng dụng thực tế.

Trong tương lai, nhóm hướng đến việc mở rộng và cải thiện mô hình bằng một số phương pháp sau:

1. *Bổ sung bộ dữ liệu*: Bên cạnh các nhãn chuẩn đoán được gán thủ công, việc tích hợp các nhãn sinh lý có nguồn gốc từ kết quả lâm sàng như chẩn đoán hình ảnh, siêu âm tim hoặc kết quả xét nghiệm máu có thể mở ra hướng tiếp cận học đa phương thức (multimodal learning) nhằm tăng cường độ chính xác và độ tin cậy của mô hình.
2. *Thử nghiệm các cấu trúc tân tiến hơn*: Việc tích hợp CAM vào SNN cũng cần được cải tiến theo hướng tăng độ chính xác vùng kích hoạt và giảm hiện tượng nhiễu. Ngoài ra, mở rộng sang các phương pháp dựa trên mô hình attention có thể giúp cung cấp những căn cứ đáng tin cậy hơn trong thực hành lâm sàng. Một hướng tiếp cận tiềm năng là phát triển các mô hình lai (hybrid) giữa CNN và SNN — ví dụ: sử dụng CNN để trích xuất đặc trưng không gian và SNN để xử lý thông tin thời gian, từ đó khai thác ưu điểm của cả hai mô hình. Đây có thể là một bước tiến đáng kể trong việc xây dựng hệ thống chẩn đoán ECG tự động vừa chính xác, vừa tiết kiệm năng lượng và dễ triển khai.
3. *Triển khai mô hình vào thực tế*: Cuối cùng, nhóm dự định phát triển một ứng dụng hoặc hệ thống nhận diện các vấn đề về tim theo thời gian thực, cũng như



tích hợp với các thiết bị đeo tay có các cảm biến để thực hiện việc đo tín hiệu về nhịp tim. Đây cũng là một trong các lý do để nhóm hướng tới xây dựng một mô hình gọn nhẹ và tiết kiệm tài nguyên.

## 6. Kết luận

Trong nghiên cứu này, nhóm đã tiến hành so sánh hai kiến trúc mạng nơ-ron sâu – CNN truyền thống và SNN tích hợp cơ chế chú ý theo kênh (CAM) – trong tác vụ phân loại tín hiệu điện tâm đồ (ECG) sử dụng bộ dữ liệu PTB-XL. Kết quả cho thấy rằng mặc dù CNN đạt hiệu năng cao trong điều kiện không bị ràng buộc về tài nguyên, thì SNN lại nổi bật với khả năng tiết kiệm năng lượng và tài nguyên tính toán nhờ cơ chế xử lý theo xung, phù hợp hơn với các ứng dụng y tế di động và thiết bị nhúng.

Việc tích hợp CAM vào SNN góp phần cải thiện khả năng học đặc trưng của mô hình, đồng thời tăng tính giải thích – yếu tố quan trọng trong các hệ thống hỗ trợ chẩn đoán lâm sàng. Bên cạnh đó, việc huấn luyện SNN trực tiếp bằng gradient xấp xỉ cho thấy tiềm năng thay thế các phương pháp chuyển đổi mô hình phức tạp mà vẫn đảm bảo hiệu quả học sâu.

Tổng thể, kết quả thực nghiệm cho thấy rằng SNN+CAM là một hướng tiếp cận đầy hứa hẹn để phát triển các hệ thống phân tích tín hiệu ECG thông minh, thân thiện với năng lượng và có khả năng mở rộng trong môi trường thực tế. Trong tương lai, các nghiên cứu có thể tập trung mở rộng mô hình sang các dạng tín hiệu sinh lý khác, tối ưu hóa cấu trúc mạng SNN, cũng như triển khai thực nghiệm trên phần cứng neuromorphic để đánh giá toàn diện hơn hiệu quả triển khai.

## Tài liệu

- [1] J. G. Webster, *Medical Instrumentation: Application and Design*, 4th ed. Hoboken, NJ, USA: Wiley, 2009.
- [2] U. R. Acharya *et al.*, “A deep convolutional neural network model to classify heartbeats,” *Comput. Biol. Med.*, vol. 89, pp. 389–396, 2017.
- [3] A. Y. Hannun *et al.*, “Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network,” *Nat. Med.*, vol. 25, no. 1, pp. 65–69, 2019.
- [4] K. Roy, A. Jaiswal, and P. Panda, “Towards spike-based machine intelligence with neuromorphic computing,” *Nature*, vol. 575, no. 7784, pp. 607–617, 2019.
- [5] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. Maida, “Deep learning in spiking neural networks,” *Neural Netw.*, vol. 111, pp. 47–63, 2019.
- [6] S. Śmigiel, K. Pałczyński, and D. Ledziński, “ECG signal classification using deep learning techniques based on the PTB-XL dataset,” *Entropy*, vol. 23, no. 1, 2021.
- [7] P. Mach and Z. Becvar, “Mobile edge computing: A survey on architecture and computation offloading,” *IEEE Commun. Surv. Tutor.*, vol. 19, no. 3, pp. 1628–1656, 2017.
- [8] P. Plagwitz, F. Hannig, J. Teich, and O. Keszocze, “To spike or not to spike? A quantitative comparison of SNN and CNN FPGA implementations,” *ACM Trans. Embedd. Comput. Syst.*, vol. 42, no. 42, Art. 42, 2023.
- [9] E. Hunsberger and C. Eliasmith, “Spiking deep networks with LIF neurons,” 2015. [Online]. Available: <https://arxiv.org/abs/1510.08829>
- [10] J. H. Lee, T. Delbruck, and M. Pfeiffer, “Training deep spiking neural networks using backpropagation,” *Front. Neurosci.*, vol. 10, p. 508, 2016.
- [11] P. Panda and K. Roy, “Unsupervised regenerative learning of hierarchical features in spiking deep networks for object recognition,” in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Vancouver, BC, Canada, pp. 299–306, 2016.
- [12] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. Maida, “Deep learning in spiking neural networks,” *Neural Netw.*, vol. 111, pp. 47–63, 2019.
- [13] Y. Xing *et al.*, “Accurate ECG classification based on spiking neural network and attentional mechanism for real-time implementation on personal portable devices,” *Electronics*, vol. 11, no. 12, p. 1889, 2022.
- [14] D. Wu, X. Yi, and X. Huang, “A little energy goes a long way: Build an energy-efficient, accurate spiking neural network from convolutional neural network,” *Front. Neurosci.*, vol. 16, 2022.
- [15] S. Lu and F. Xu, “Linear leaky-integrate-and-fire neuron model based spiking neural networks and its mapping relationship to deep neural networks,” *Front. Neurosci.*, vol. 16, 2022.
- [16] P. Wagner, N. Strodthoff, R. Bousseljot, W. Samek, and T. Schaeffter, “PTB-XL, a large publicly available electrocardiography dataset (version 1.0.3),” *PhysioNet*, 2022. [Online]. Available: <https://physionet.org/content/ptb-xl/1.0.3/>
- [17] P. Wagner *et al.*, “PTB-XL: A large publicly available ECG dataset,” *Sci. Data*, vol. 7, no. 1, pp. 1–15, 2020.
- [18] A. Wagner, N. Strodthoff, T. D. Bousseljot, R. Kreiseler, F. Samek, and T. Schaeffter, “PTB-XL, a large publicly available electrocardiography dataset,” *Scientific Data*, vol. 7, no. 1, pp. 1–15, 2020.
- [19] H. I. Fawaz, B. Lucas, G. Forestier, C. Pelletier, D. F. Schmidt, J. Weber, et al. “InceptionTime: Finding AlexNet for time series classification,” 2019. [Online]. Available: <https://arxiv.org/abs/1909.04939>
- [20] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [21] L. D. Sharma and R. K. Sunkaria, “Inferior myocardial infarction detection using stationary wavelet transform and machine learning approach,” *Signal, Image and Video Processing*, vol. 12, no. 2, pp. 199–206, 2017.
- [22] Z. Yan, J. Zhou, and W.-F. Wong, “Energy efficient ECG classification with spiking neural network,” *Biomed. Signal Process. Control*, vol. 63, p. 102170, 2021.
- [23] Z. Yan, Z. Bai, and W.-F. Wong, “Reconsidering the energy efficiency of spiking neural networks,” 2024. [Online]. Available: <https://arxiv.org/abs/2409.08290>