

NHẬN DIỆN DẤU HIỆU NHỊP TIM BẤT THƯỜNG DỰA TRÊN VIDEO MRI SỬ DỤNG PHƯƠNG PHÁP 3D CNN VÀ LSTM

ĐỀ CƯƠNG LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC

Đặng Lâm Tùng – 1713856
Giảng viên hướng dẫn
TS. Phạm Quang Thái



ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA ĐIỆN – ĐIỆN TỬ, BỘ MÔN VIỄN THÔNG

Số: _____/BKĐT

Khoa: **Điện – Điện tử**

Bộ môn: **Viễn thông**

NHIỆM VỤ LUẬN VĂN TỐT NGHIỆP

- Họ và tên: Đặng Lâm Tùng, MSSV: 1713856
- Ngành: Điện – Điện tử, Chuyên ngành: Kỹ thuật Điện tử – Truyền thông
- Đề tài: Nhận diện dấu hiệu nhịp tim bất thường dựa trên video MRI sử dụng phương pháp 3D CNN và LSTM
- Nhiệm vụ:
 - Nhiệm vụ 1
 - Nhiệm vụ 2
 - Nhiệm vụ 3
- Ngày giao nhiệm vụ: 2-10-2019
- Ngày hoàn thành nhiệm vụ: 2-10-2020
- Họ và tên người hướng dẫn: TS. Phạm Quang Thái
BM Viễn Thông, Khoa Điện – Điện Tử

Phần hướng dẫn:
100%

Nội dung và yêu cầu LVTN đã được thông qua Bộ Môn.

Tp. HCM, Ngày __ tháng __ năm 20__
CHỦ NHIỆM BỘ MÔN

Tp. HCM, Ngày __ tháng __ năm 20__
NGƯỜI HƯỚNG DẪN CHÍNH

PGS. TS. Hà Hoàng Kha

TS. Phạm Quang Thái

PHẦN DÀNH CHO KHOA, BỘ MÔN:

Người duyệt (chấm sơ bộ): _____

Đơn vị: _____

Ngày bảo vệ: _____

Điểm tổng kết: _____

Nơi lưu trữ luận văn: _____

LỜI CẢM ƠN

Sinh viên viết phần lời cảm ơn vào đây. Lưu ý giới hạn không quá 1 trang.

Tp. HCM, Ngày 13 tháng 1 năm 2021 Đặng

Lâm Tùng

LỜI CAM ĐOAN

Tôi tên: Đặng Lâm Tùng (MSSV: 1713856), là sinh viên chuyên ngành Kỹ thuật Điện tử - Truyền thông, tại Trường Đại học Bách Khoa, Đại học Quốc gia thành phố Hồ Chí Minh. Tôi xin cam đoan những nội dung sau đều là sự thật:

- Công trình nghiên cứu này hoàn toàn do chính tôi thực hiện;
- Các tài liệu và trích dẫn trong luận văn này được tham khảo từ các nguồn thực tế, có uy tín và độ chính xác cao;
- Các số liệu và kết quả của công trình này được tôi tự thực hiện một cách độc lập và trung thực.

Tp. HCM, Ngày 13 tháng 1 năm 2021 Đặng

Lâm Tùng

TÓM TẮT NỘI DUNG

Trong cuộc sống hiện đại ngày nay, chất lượng cuộc sống ngày càng được nâng cao, thì vấn đề sức khỏe, đặc biệt là về các bệnh tim mạch ngày càng được quan tâm, những phương pháp mới tiến bộ hơn được sử dụng vào trong chẩn đoán và điều trị. Một phương pháp chẩn đoán hiện nay được quan tâm nghiên cứu rất nhiều đó là phương pháp chụp cắt lớp MRI. Khi chụp MRI khu vực tim sẽ được chụp nhiều lát cắt (slide), số slide này có số lượng tầm khoảng 8 - 16 nhất cắt, và để chẩn đoán thì cần phải chụp nhiều chuỗi ảnh như vậy để có thể quan sát quy trình tim đập. Để xử lý loại dữ liệu này, em sử dụng phương pháp 3D CNN kết hợp với LSTM để phân loại các chuỗi ảnh này. Đề cương này em đã cài đặt một thuật toán và train nó trên một tập dataset chuỗi ảnh MRI tim nhỏ. Thuật toán này có thể phân loại chuỗi ảnh tim là bình thường hay không với độ chính xác 98.4 % sau 10 epochs

ABSTRACT

Tóm tắt luận văn bằng tiếng Anh. Giới hạn trong 1 trang. Nội dung tóm tắt tham khảo phần hướng dẫn và ví dụ của tóm tắt tiếng Việt.

MỤC LỤC

1	Giới thiệu	1
1.1	Đặt vấn đề	1
1.2	Phạm vi và phương pháp nghiên cứu	1
2	Cơ sở lý thuyết	3
2.1	Lý thuyết về xử lý ảnh:	3
2.1.1	Mạng Neuron tích chập (CNN):	3
2.2	Long sort term memory	6
2.2.1	Recurrent Neural Network	6
2.2.2	Long Short Term Memory	7
2.2.3	Cách sử dụng Latex cho báo cáo luận văn	10
2.3	Cách viết chương cơ sở lý thuyết	10
2.3.1	Nội dung chính	10
2.3.2	Cách viết tắt	11
2.3.3	Cách viết công thức toán học	11
2.3.4	Cách chèn hình vẽ	12
2.3.5	Cách thêm và sử dụng tài liệu tham khảo	12
2.4	Kết luận chương	14
3	Kết quả nghiên cứu	15
3.1	Phương pháp tiếp cận	15
3.2	Kết quả và phân tích	16
3.3	Kết luận chương	18
4	Kết luận	19
4.1	Tóm tắt và kết luận chung	19
4.2	Hướng phát triển	19
4.3	Mục tiêu và tiến độ thực hiện	19
A	Code chương trình	23

DANH SÁCH HÌNH VẼ

Hình 2.1	Công thức tích chập	3
Hình 2.2	Biểu diễn tích chập	4
Hình 2.3	Biểu diễn tích chập	4
Hình 2.4	Biểu diễn tích chập	5
Hình 2.5	Biểu diễn tích chập	6
Hình 2.6	RNN unrolled	7
Hình 2.7	Biểu diễn RNN theo thời gian (thứ tự trong sequence)	7
Hình 2.8	Long term memory ảnh hưởng đến RNN	7
Hình 2.9	Biểu diễn RNN theo thời gian (thứ tự trong sequence)	8
Hình 2.10	Ví dụ 1 cell của LSTM	8
Hình 2.11	Ví dụ 1 cell của LSTM	8
Hình 2.12	Ví dụ 1 cell của LSTM	9
Hình 2.13	Ví dụ 1 cell của LSTM	9
Hình 2.14	Ví dụ 1 cell của LSTM	9
Hình 2.15	Ví dụ 1 cell của LSTM	9
Hình 2.16	Phổ tín hiệu Orthogonal Frequency Division Multiplexing (OFDM) .	13
Hình 3.1	Lưu đồ giải thuật phía phát của giao thức Go back N	16
Hình 3.2	Sơ đồ thí nghiệm hệ thống Visible Light Communication (VLC) . . .	17
Hình 3.3	Đặc trưng nước đường từ 12 °Brix đến 16 °Brix	17

DANH SÁCH BẢNG

Bảng 3.1	Kết quả xác suất phân loại nhãn bằng Convolutional Neural Networks (CNN)	18
Bảng 4.1	Tiến độ thực hiện luận văn	19

DANH SÁCH TỪ VIẾT TẮT

CNN Convolutional Neural Networks

IEEE Institute of Electrical and Electronics Engineers

OFDM Orthogonal Frequency Division Multiplexing

RMSE Root Mean Square Error

SNR Signal to Noise Ratio

VLC Visible Light Communication

Chương 1. GIỚI THIỆU

1.1 Đặt vấn đề

Trong cuộc sống hiện đại ngày nay, chất lượng cuộc sống ngày càng được nâng cao, thì vấn đề sức khỏe, đặc biệt là về các bệnh tim mạch ngày càng được quan tâm, những phương pháp mới tiến bộ hơn được sử dụng vào trong chẩn đoán và điều trị. Một phương pháp chẩn đoán hiện nay được quan tâm nghiên cứu rất nhiều đó là phương pháp chụp cắt lớp MRI.

Thông thường, khi chụp MRI khu vực tim sẽ được chụp nhiều lát cắt (slide), số slide này có số lượng tầm khoảng 8 - 16 nhất cắt, và để chẩn đoán thì cần phải chụp nhiều chuỗi ảnh như vậy để có thể quan sát quy trình tim đập. Data này gồm một số video là video chụp MRI tim trong thời gian 1s - 30 khung hình, gồm 2 class là normal và abnormal, chất lượng video 4K, dung lượng > 700 MB/file, được cung cấp bởi CLB AI đại học Bách Khoa, do thầy Quản Thành Thơ, khoa khoa học máy tính quản lý.

Em hiểu rõ bản thân không có nhiều kiến thức về lĩnh vực này, nên em chỉ coi bài toán này như một bài toán về deep learning, hoàn toàn chưa có đủ giá trị thực hiện như một mô hình chẩn đoán hoàn chỉnh có thể áp dụng được trong thực tế.

Nhiệm vụ của đề cương này là:

- + Tìm hiểu phương pháp xử lý hiệu quả với bài toán, bài toán này là bài toán phân loại (classification) một video MRI.
- + Cài đặt thuật toán xử lý bài toán, thực hiện kiểm chứng hiệu quả của phương pháp.

1.2 Phạm vi và phương pháp nghiên cứu

Trong mục này, sinh viên liệt kê các phương pháp đã dùng để kiểm nghiệm giả thiết trong bài toán nghiên cứu (chẳng hạn mô phỏng hay thực nghiệm). Đồng thời, sinh viên liệt kê các giới hạn trong quá trình thực hiện các phương pháp (chẳng hạn mô phỏng trong điều kiện gì, đo đạc trong điều kiện gì).

Phạm vi và phương pháp nghiên cứu trong đề cương luận văn như sau:

- Tìm hiểu dữ liệu bài toán,
- Xây dựng mô hình nhận diện (detection) tim trong ảnh chụp MRI sử dụng Tensorflow Object Detection.
- Xây dựng mô hình phân loại (classification) chuỗi ảnh tim (video) sử dụng 3D convolution và LSTM,
- Thử nghiệm mô hình.

Chương 2. CƠ SỞ LÝ THUYẾT

2.1 Lý thuyết về xử lý ảnh:

2.1.1 Mạng Neuron tích chập (CNN):

Tích chập (Convolution)

Trong xử lý số tín hiệu, tích chập là một khái niệm hết sức quen thuộc, nó là một phép toán cho 2 hàm số để ra một hàm số khác phép tích chập có thể thể hiện nhiều kiểu biến đổi (có thể là lọc tín hiệu, xét xem tín hiệu có theo mẫu nào không,...)

Tích chập của hàm số f và g được viết là $f * g$, là 1 phép biến đổi tích phân đặc biệt:

$$\begin{aligned}(f * g)(t) &\stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau) g(t - \tau) d\tau \\ &= \int_{-\infty}^{\infty} f(t - \tau) g(\tau) d\tau. \quad (\text{giao hoán})\end{aligned}$$

Hình 2.1: Công thức tích chập

Công thức của tích chập thể hiện tính chất của tích chập đó là khi thực hiện, ta phải trừ ngược lại đáp ứng xung của hệ thống để tích chập thực hiện trước khi tín hiệu xuất hiện (thể hiện việc tín hiệu có thể xuất hiện ở trước điểm 0)

Trong đó ta thực hiện trừ ngược và lật hàm g lại sau đó tích phân trên cả miền của tín hiệu

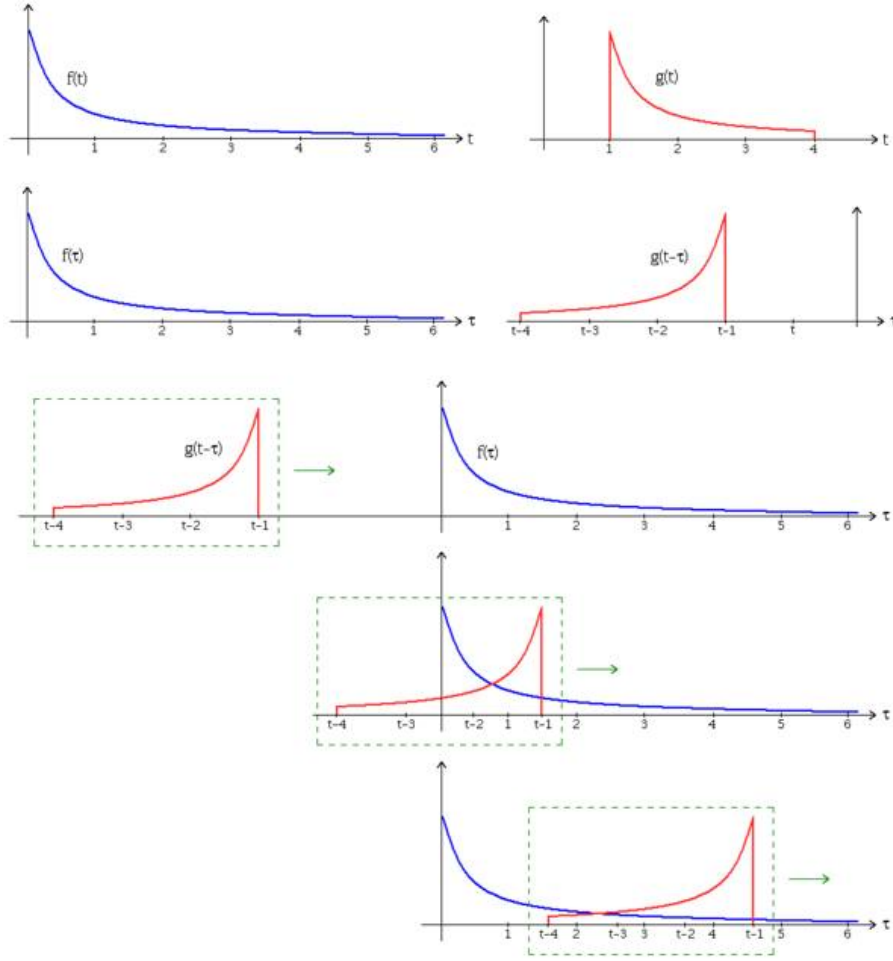
Ta hoàn toàn có thể coi đây như là tín hiệu 2 chiều vậy, chính vì vậy để lọc các tính chất của ảnh ta có thể áp dụng được các công cụ như tích chập. Chính vì thế dẫn đến các bộ lọc được dùng rất nhiều trong computer vision. Tuy nhiên ở đây tín hiệu có 2 chiều, và hơn nữa lại không có trường hợp tín hiệu xuất hiện trước $t = 0$ nên tích chập ở đây không cần phải lật kernel như tín hiệu 1 chiều. Chính vì vậy nó giống một khái niệm khác là tương quan chéo, mặc dù bản chất không giống nhau, phép này vẫn là tích chập.

Công thức của tích chập 2 chiều:

Ở đây có thể thấy phép convolution được thực hiện giống với 1 chiều, và phép tích phân được thay thế bằng phép tổng, phù hợp với tín hiệu rời rạc

Mạng neuron tích chập

Mạng Nơ-ron Tích Chập có kiến trúc khác với Mạng Nơ-ron thông thường. Mạng Nơ-ron bình thường chuyển đổi đầu vào thông qua hàng loạt các tầng ẩn. Mỗi tầng là một tập



Hình 2.2: Biểu diễn tích chập

$$(I * K)_{ij} = \sum_{m=0}^{k_1-1} \sum_{n=0}^{k_2-1} I(i-m, j-n) K(m, n) \quad (2)$$

$$= \sum_{m=0}^{k_1-1} \sum_{n=0}^{k_2-1} I(i+m, j+n) K(-m, -n) \quad (3)$$

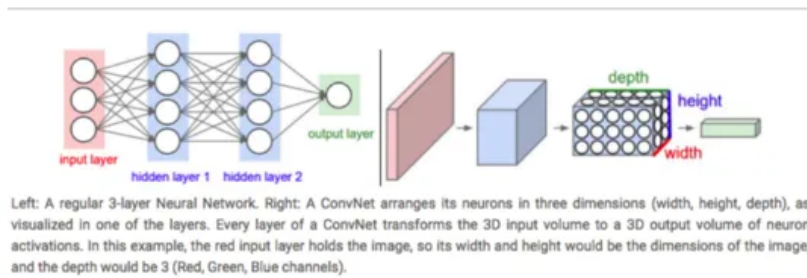
Hình 2.3: Biểu diễn tích chập

các nơ-ron và các tầng được liên kết đầy đủ với các nơ-ron ở tầng trước đó. Và ở tầng cuối cùng sẽ là tầng kết quả đại diện cho dự đoán của mạng.

Đầu tiên, mạng Nơ-ron Tích Chập được chia thành 3 chiều: rộng, cao, và sâu. Kế đến, các nơ-ron trong mạng không liên kết hoàn toàn với toàn bộ nơ-ron kế đến nhưng chỉ liên kết tới một vùng nhỏ. Cuối cùng, một tầng đầu ra được tối giản thành véc-tơ của giá trị xác suất.

CNNs gồm hai thành phần:

- Phần tầng ẩn hay phần rút trích đặc trưng: trong phần này, mạng sẽ tiến hành tính toán hàng loạt phép tích chập và phép hợp nhất (pooling) để phát hiện các đặc trưng. Ví dụ: nếu ta có hình ảnh con ngựa vằn, thì trong phần này mạng sẽ nhận diện các sọc vằn, hai tai, và bốn chân của nó.



Hình 2.4: Biểu diễn tích chập

- **Phần phân lớp:** tại phần này, một lớp với các liên kết đầy đủ sẽ đóng vai trò như một bộ phân lớp các đặc trưng đã rút trích được trước đó. Tầng này sẽ đưa ra xác suất của một đối tượng trong hình.

Tích chập là một khối quan trọng trong CNN. Thuật ngữ tích chập được dựa trên một phép hợp nhất toán học của hai hàm tạo thành hàm thứ ba. Phép toán này kết hợp hai tập thông tin khác nhau. Trong trường hợp CNN, tích chập được thực hiện trên giá trị đầu vào của dữ liệu và kernel/filter (thuật ngữ này được sử dụng khác nhau tùy tình huống) để tạo ra một bản đồ đặc trưng (feature map).

Ta thực hiện phép tích chập bằng cách trượt kernel/filter theo dữ liệu đầu vào. Tại mỗi vị trí, ta tiến hành phép nhân ma trận và tính tổng các giá trị để đưa vào bản đồ đặc trưng.

Trong hình dưới đây, thành phần kernel/filter (màu xanh lá) trượt trên đầu vào (màu xanh dương) và kết quả được trả về bản đồ đặc trưng (màu đỏ). Kernel/filter có kích thước là 3×3 trong ví dụ này.

Trong thực tế, tích chập được thực hiện trên không gian 3 chiều. Vì mỗi hình ảnh được biểu diễn dưới dạng 3 chiều: rộng, cao, và sâu. Chiều sâu ở đây chính là giá trị màu sắc của hình (RGB). Ta thực hiện phép tích chập trên đầu vào nhiều lần khác nhau. Mỗi lần sử dụng một kernel/filter khác nhau. Kết quả ta sẽ thu được những bản đồ đặc trưng khác nhau. Cuối cùng, ta kết hợp toàn bộ bản đồ đặc trưng này thành kết quả cuối cùng của tầng tích chập.

Tương tự như mạng nơ-ron thông thường, ta sử dụng một hàm kích hoạt (activate function) để có đầu ra dưới dạng phi tuyến. Trong trường hợp CNN, đầu ra của phép tích chập sẽ đi qua hàm kích hoạt nào đó ví dụ như hàm ReLU (rectified linear units).

Trong quá trình trượt kernel/filter trên dữ liệu đầu vào, ta sẽ quy định một bước nhảy (stride) với mỗi lần di chuyển. Thông thường ta lựa chọn thường chọn bước nhảy là 1. Nếu kích thước bước nhảy tăng, kernel/filter sẽ có ít ô trùng lặp.

Bởi vì kích thước đầu ra luôn nhỏ hơn đầu vào nên ta cần một phép xử lý đầu vào để đầu ra không bị co giãn. Đơn giản ta chỉ cần thêm một lề nhỏ vào đầu vào. Một lề với giá trị 0 sẽ được thêm vào xung quanh đầu vào trước khi thực hiện phép tích chập.

Thông thường, sau mỗi tầng tích chập, ta sẽ cho kết quả đi qua một tầng hợp nhất (pooling layer). Mục đích của tầng này là để nhanh chóng giảm số chiều. Việc này giúp giảm thời gian học và hạn chế việc overfitting.

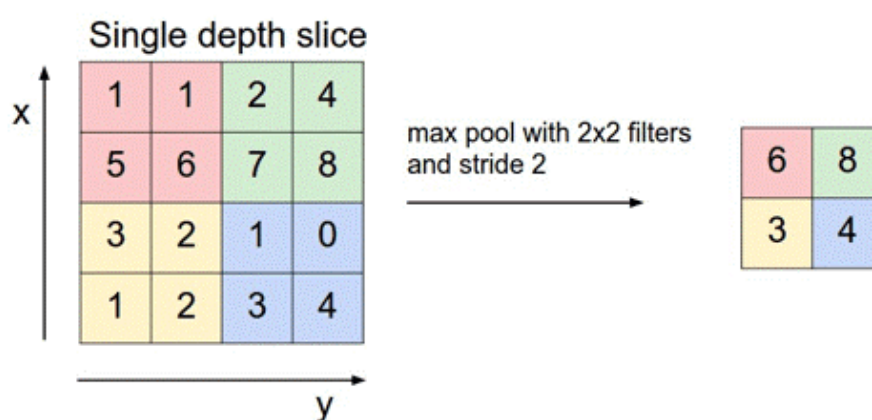
Một phép hợp nhất đơn giản thường được dùng đó là max pooling, phép này lấy giá trị lớn nhất của một vùng để đại diện cho vùng đó. Kích thước của vùng sẽ được xác định trước để giảm kích thước của bản đồ đặc trưng nhanh chóng nhưng vẫn giữ được thông tin cần thiết.

Tổng kết lại khi sử dụng CNN, ta cần chú ý đến 4 tham số quan trọng: • Kích thước kernel/filter

- Số lượng kernel/filter
- Kích thước bước nhảy (stride)
- Kích thước lề (padding)

Phân lớp

Trong phần phân lớp, ta sử dụng một vài tầng với kết nối đầy đủ để xử lý kết quả của phân tích chập. Vì đầu vào của mạng liên kết đầy đủ là 1 chiều, ta cần làm phẳng đầu vào trước khi phân lớp. Tầng cuối cùng trong mạng CNN là một tầng liên kết đầy đủ, phần này hoạt động tương tự như mạng nơ-ron thông thường.



Hình 2.5: Biểu diễn tích chập

Kết quả thu được cuối cùng cũng sẽ là một véc-tơ với các giá trị xác suất cho việc dự đoán như mạng nơ-ron thông thường.

Cài đặt Latex bao gồm hai phần là tập lệnh chính (Miktex) có trang chủ tại đây: <https://miktex.org/> và phần mềm giao diện với người dùng. Phần mềm giao diện có rất nhiều, tác giả báo cáo mẫu này dùng TexStudio (<http://www.texstudio.org/>) do là phần mềm miễn phí.

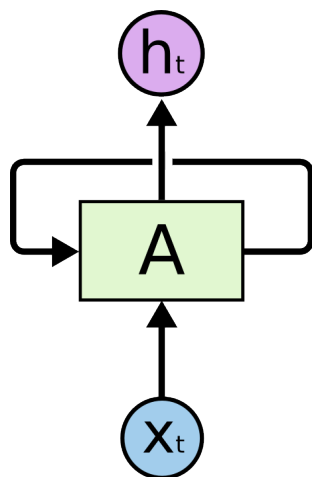
2.2 Long sort term memory

2.2.1 Recurrent Neural Network

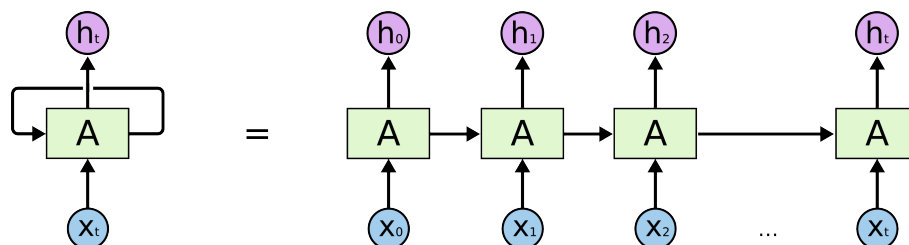
Để thực hiện xử lý các giá trị có tính chất chuỗi (sequence) với mạng nơ-ron, kiến trúc Recurrent Neural Network được sử dụng. Mô hình này có cấu trúc như hình dưới:

Hình trên là biểu diễn ý tưởng cơ bản của mạng nơ-ron RNN: dữ liệu từ các thời điểm khác nhau trong thời gian đều có ảnh hưởng đến dữ liệu ở thời điểm tiếp sau:

Bài blog rất nổi tiếng của Andrew Karpathy đã nói về tính hiệu quả của RNN, tuy nhiên RNN có một số vấn đề, nên thực tế biến thể của nó là LSTM được sử dụng nhiều hơn.

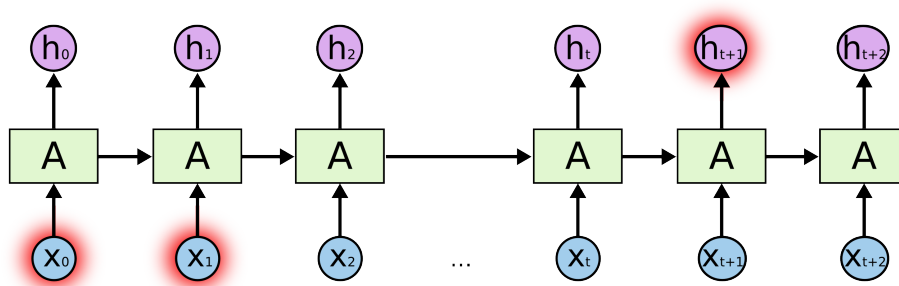


Hình 2.6: RNN unrolled



Hình 2.7: Biểu diễn RNN theo thời gian (thứ tự trong sequence)

Ví dụ trong trường hợp chuỗi dữ liệu cho RNN quá dài, những dữ liệu ban đầu cũng ảnh hưởng đến đầu ra cho dù những dữ liệu ấy có thể không còn liên quan đến dữ liệu hiện tại:

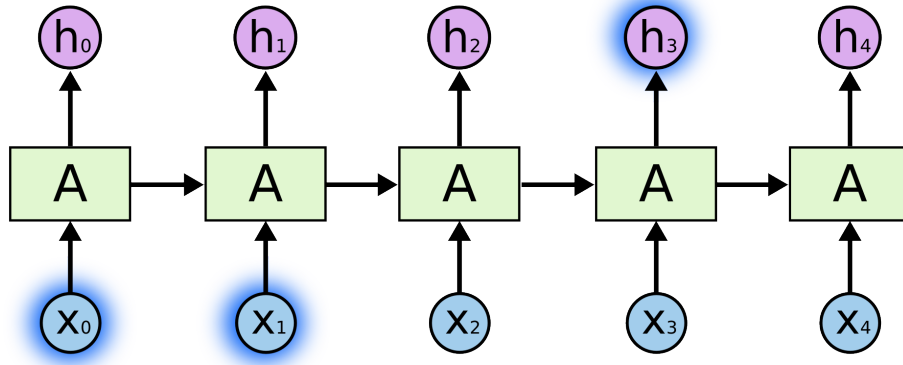


Hình 2.8: Long term memory ảnh hưởng đến RNN

Trường hợp chúng ta mong muốn đó là dữ liệu đầu vào của RNN chỉ ảnh hưởng đến đầu ra ở một thời điểm xác định như sau:

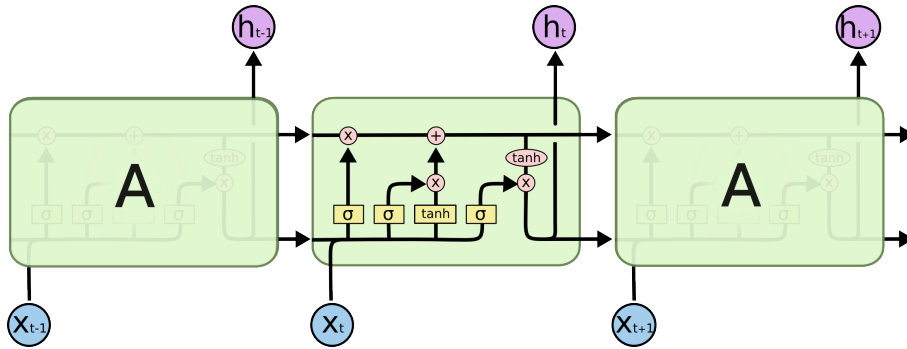
2.2.2 Long Short Term Memory

LSTM là một biến thể của RNN, trong đó các mạng nơ-ron không chỉ sử dụng dữ liệu từ các điểm dữ liệu trước mà còn có cổng "quên" để các giá trị quá xa trong quá khứ không

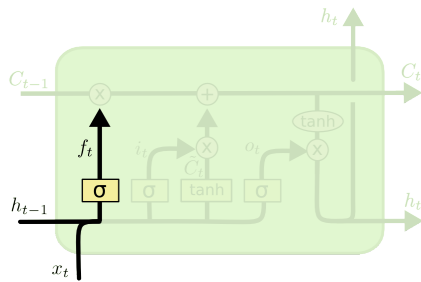


Hình 2.9: Biểu diễn RNN theo thời gian (thứ tự trong sequence)

ảnh hưởng đến các giá trị hiện tại.



Hình 2.10: Ví dụ 1 cell của LSTM



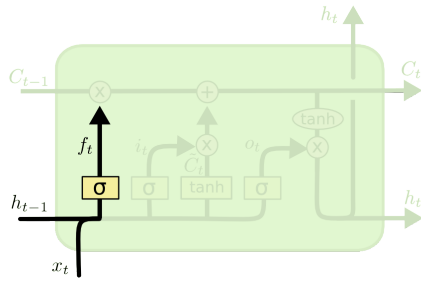
$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Hình 2.11: Ví dụ 1 cell của LSTM

Hình dưới thể hiện cổng "quên" của 1 cell LSTM, đây là một lớp nơ-ron sigmoid có input là đầu ra của cell trước đó h_{t-1} và trọng số W_f .

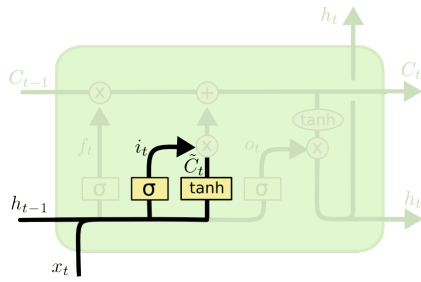
Cổng quên này có thể có giá trị giữa 0 và 1, 0 tức cell sau hoàn toàn "quên" các giá trị từ cell trước, 1 tức là cell sau nhận toàn bộ giá trị của cell trước.

Tiếp theo ta có cổng input, giá trị trạng thái tiềm năng \hat{C}_t của cell sẽ được cập nhật theo công thức như hình, giá trị này được tính bằng một lớp nơ-ron tanh với input là giá trị đầu vào h_{t-1} và giá trị input của cell hiện tại là x_t , đầu ra của cổng này được nhân pointwise với giá trị của output cell trước với một lớp sigmoid.



$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

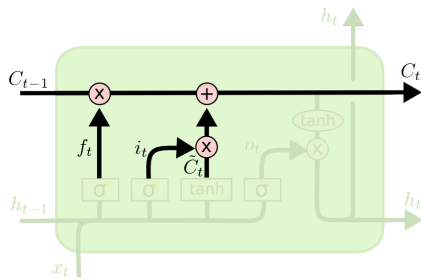
Hình 2.12: Ví dụ 1 cell của LSTM



$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

Hình 2.13: Ví dụ 1 cell của LSTM

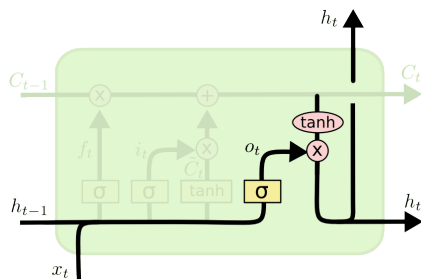


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Hình 2.14: Ví dụ 1 cell của LSTM

Tiếp tục với bước cập nhật giá trị \mathbf{C}_t theo công thức như hình trên, ta nhận giá trị $\hat{\mathbf{C}}_t$ với giá trị quên đã tính ở trước và cộng với giá trị

Cuối cùng, cổng output của cell như sau:



$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

Hình 2.15: Ví dụ 1 cell của LSTM

Giá trị output \mathbf{h}_t được tính dựa trên trạng thái của cell \mathbf{C}_t sau khi qua một lớp mạng

ơ-rơ tanh (để đưa giá trị về đoạn - 1 đến 1), giá trị này được nhân với giá trị đầu ra của lớp trước sau khi qua một lớp sigmoid.

2.2.3 Cách sử dụng Latex cho báo cáo luận văn

Các sinh viên muốn đọc hiểu thêm về cách sử dụng Latex căn bản có thể tham khảo tại đây: <https://tobi.oetiker.ch/lshort/lshort.pdf>. Các sinh viên chỉ muốn hiểu đủ để viết báo cáo luận văn, có thể đọc trong báo cáo này.

Việc sinh viên cần làm là viết các nội dung vào các file tương ứng trong thư mục con “Text” đi kèm với thư mục chính này. Cụ thể, sinh viên viết lời cảm ơn vào file “loicamon.tex”, viết tóm tắt tiếng Việt vào file “tomtatviet.tex”, viết tóm tắt tiếng Anh vào file “tomtateng.tex”, và viết nội dung các chương vào các file tương ứng.

Sau đó, sinh viên mở file “thesis.tex” trong thư mục chính. Trong file này, các dòng lệnh `\newcommand` là định nghĩa các biến số.

Ví dụ 2.2.1 `\newcommand\studenta{Nguyễn Văn A}`: định nghĩa biến tên “studenta” có giá trị là “Nguyễn Văn A”.

Sinh viên chỉnh các biến số này cho phù hợp với họ tên, MSSV, tên luận văn, giảng viên hướng dẫn, ngày giao nhiệm vụ luận văn, ngày hoàn thành nhiệm vụ luận văn, và nội dung nhiệm vụ luận văn. Nếu có 2 sinh viên làm cùng một đề tài, cần tìm các dòng có biến số “studentb” và “mssvb” và bỏ dấu % phía trước dòng tương ứng đó. Dấu % nghĩa là dòng này là dòng comment, tương tự trong các ngôn ngữ lập trình.

Sau cùng, sinh viên chạy biên dịch file “thesis.tex” là xong. Dàn trang này đã tối ưu hóa cho việc in 2 mặt báo cáo. Sinh viên chỉ cần in 2 mặt toàn bộ file “thesis.pdf” là có thể đóng tập hoàn chỉnh.

2.3 Cách viết chương cơ sở lý thuyết

2.3.1 Nội dung chính

Trong chương này, cơ sở lý thuyết về các vấn đề có liên quan như mô hình toán học, giải thuật, phương pháp, và các công trình đã công bố v.v. sẽ được trình bày. Do đó, sinh viên cần nắm cách sử dụng tài liệu tham khảo, cách viết công thức toán học, và cách trình bày hình ảnh cũng như sơ đồ.

Ví dụ 2.3.1 Một luận văn làm về hệ thống truyền dẫn quang không dây dùng ánh sáng khả kiến, *Visible Light Communication (VLC)*. Trong đó, sinh viên dùng mạng neuron tích chập, *Convolutional Neural Networks (CNN)*, để cân bằng miền tần số cho tín hiệu ghép kênh phân chia theo tần số trực giao, *Orthogonal Frequency Division Multiplexing (OFDM)*. Các mục cần trong chương này là:

- Sơ đồ khối và phương pháp hoạt động của VLC,
- Sơ đồ khối và giải thuật CNN,
- Khái niệm về cân bằng miền tần số,
- Sơ đồ khối và phương pháp điều chế / giải điều chế OFDM,

- Nội dung một số bài báo tham khảo chính để thực hiện đề tài (nếu có),
- Kết luận chương.

Cần lưu ý là trong báo cáo chỉ được phép có tối đa 3 mức chỉ mục, chẳng hạn có 1, 1.1, 1.1.1, nhưng không có 1.1.1.1. Do đó, sinh viên cần có phân bố nội dung cho phù hợp. Chỉ mục lớn nhất là lệnh `\chapter`, chỉ mục thứ hai là lệnh `\section`, và chỉ mục thứ ba là lệnh `\subsection`.

Khi tạo chỉ mục, cần có ít nhất hai chỉ mục đồng cấp, chẳng hạn có 1.1.1 và 1.1.2. Nếu chỉ có một chỉ mục, chẳng hạn chỉ có 1.1.1, thì không cần tạo chỉ mục.

2.3.2 Cách viết tắt

Nếu muốn viết tắt một cụm danh từ dài (chẳng hạn như tên một phương pháp tính toán), thì cần định nghĩa từ viết tắt ở lần đầu tiên cụm danh từ đó xuất hiện trong báo cáo. Những lần sau khi cụm danh từ này xuất hiện ta chỉ cần viết bằng từ viết tắt. Ngoài ra, danh sách từ viết tắt cần được đặt ở trước phần nội dung các chương và cần được xếp theo thứ tự chữ cái.

Do đó, nếu tạo danh sách này theo cách thủ công, việc xóa hay thay đổi vị trí các đoạn văn trong quá trình chỉnh sửa có thể làm thay đổi danh sách cũng như cách viết mà không kiểm tra được.

Sinh viên có thể dùng thư viện “glossary” của Latex để tạo danh sách từ viết tắt. Tuy nhiên, thư viện này đòi hỏi cài đặt ngôn ngữ Perl. Bản template này dùng thư viện “acro”, vốn không linh hoạt như thư viện “glossary” nhưng không đòi hỏi cài đặt thêm.

Sinh viên có thể tham khảo cách định nghĩa một từ viết tắt trong file “tuviettat.tex” trong thư mục “Text”. Trong đó, mỗi từ viết tắt có một nhãn, tên tắt và tên đầy đủ. Mỗi lần cần viết tắt ta dùng câu lệnh `\ac{nhãn}`. Chẳng hạn, lệnh `\ac{snr}` để viết tắt Signal to Noise Ratio (SNR). Chương trình tự động quyết định cách format từ viết tắt cũng như tạo danh sách đúng chuẩn.

2.3.3 Cách viết công thức toán học

Các sinh viên thường copy công thức toán học dạng hình ảnh và dán vào file báo cáo để tiết kiệm thời gian. Việc này có hai bất lợi là tên biến số trong cùng một công thức nhưng copy từ hai nguồn khác nhau có thể khác nhau dẫn đến công thức không có ý nghĩa, và format font chữ của các công thức toán copy từ các nguồn khác nhau là khác nhau. Do đó, khi viết luận văn các sinh viên không được viết công thức toán bằng cách copy dạng hình ảnh này.

Công thức toán là điểm mạnh của Latex. Xem file “chuong2.tex” để thấy cách viết các công thức toán học sau.

Ví dụ đầu tiên là các công thức giải tích mà các sinh viên làm về lớp vật lý và hệ thống analog thường gặp:

$$\oint_S \mathbf{D} \cdot d\mathbf{S} = \int_V \rho dV. \quad (2.1)$$

Đây là công thức định luật Gauss cho trường điện. Trong lệnh này, `\oint` nghĩa là tích phân (integral) toàn mặt kín. `_S` nghĩa là tên mặt kín S nằm dưới dấu tích phân. `\mathbf{D}` nghĩa là biến \mathbf{D} viết đậm do là ký hiệu vector. `\cdot` nghĩa là dấu tích vô hướng vector. `\mathrm{d}` nghĩa là chữ d là vi phân viết thường. `\rho` nghĩa là biến chữ cái Hy Lạp ρ . V nghĩa là chữ V viết ở dạng biến số in nghiêng.

Khi viết công thức trong Latex, công thức đã được tự động đánh số theo đúng chuẩn. Lệnh `\label{eq:ten-cong-thuc}` dùng để đặt nhãn cho công thức và tham khảo đến sau này bằng lệnh `\eqref{eq:ten-cong-thuc}`.

Lưu ý khi viết công thức cần giải thích rõ các biến số và đơn vị (nếu có) trong công thức. Chẳng hạn, trong công thức (2.1), \mathbf{D} (C/m^2) là vector mật độ điện thông đi qua mặt kín S (m^2), và ρ (C/m^3) là mật độ điện tích trong thể tích V (m^3) nằm trong mặt kín S . Khi đọc bản thảo trong file “chuong2.tex”, sinh viên sẽ thấy các dấu “\$” trong câu trên nhằm viết phần nằm giữa chúng theo định dạng công thức toán học. Chẳng hạn viết $\$V\sim 2\$$ sẽ là V^2 .

Một ví dụ khác sử dụng tổng thay vì tích phân là:

$$RMSE = \frac{1}{m} \sum_{j=1}^m (\hat{\mathbf{c}} - \mathbf{c})^2. \quad (2.2)$$

Công thức (2.2) là công thức ước lượng sai số trung bình bình phương, Root Mean Square Error (RMSE), giữa vector thực \mathbf{c} và vector ước lượng $\hat{\mathbf{c}}$ sau m lần đo đạc. Đây là công thức thường gặp trong các bài toán xử lý tín hiệu.

Một ví dụ dạng ma trận cho các lĩnh vực liên quan đến đại số tuyến tính là:

$$\mathbf{S} = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} = \begin{bmatrix} 0.7\angle -110^\circ & 0.08\angle 60^\circ \\ 4\angle 60^\circ & 0.6\angle -70^\circ \end{bmatrix}. \quad (2.3)$$

Trong công thức (2.3), ma trận \mathbf{S} là ma trận tán xạ của một mạng hai cửa siêu cao tần.

Tùy vào yêu cầu của báo cáo, sinh viên có thể tự tìm các cách viết công thức khác trên mạng Internet.

2.3.4 Cách chèn hình vẽ

Trong chương 2, sinh viên thường dùng các hình vẽ có sẵn với các định dạng phổ biến như .jpg và .png. Để chèn một hình (chẳng hạn tên là “ofdm.png”) vào báo cáo, sinh viên copy hình tương ứng vào thư mục “Figure” và tham khảo đoạn code trong mục 2.3.4 để chèn hình ảnh.

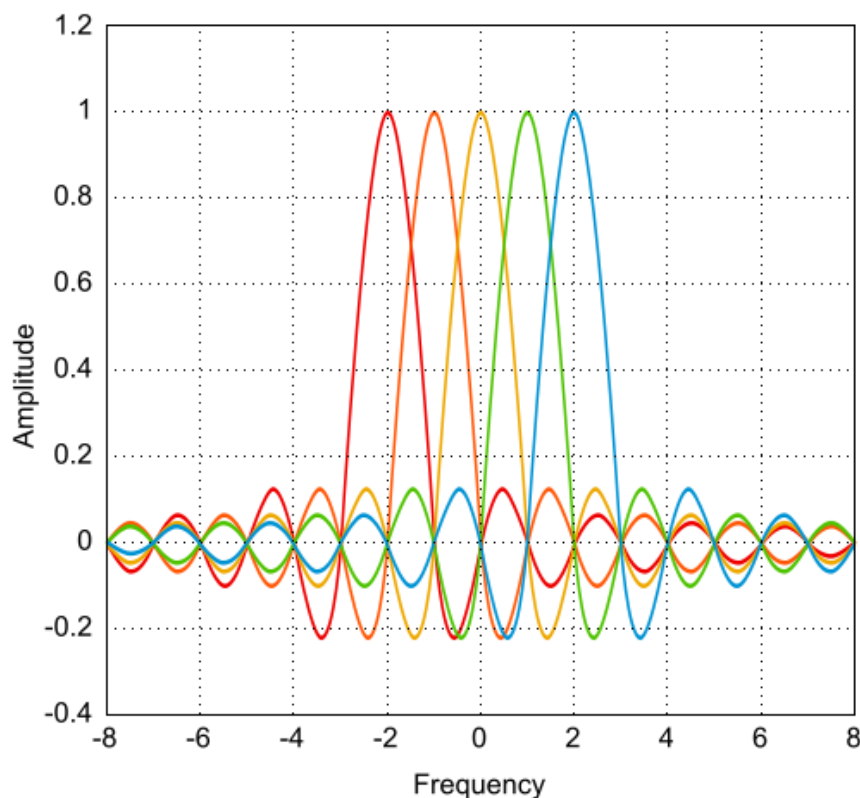
Sinh viên cần lưu ý rằng, khác với Word, Latex sẽ tự tìm vị trí tốt nhất để chèn hình ảnh này vào văn bản khi dàn trang. Hình ảnh có thể nằm phía trên, phía dưới, thậm chí nằm khác trang với đoạn văn miêu tả về hình ảnh. Do đó, tránh dùng cách miêu tả như “trong hình dưới, ta thấy phổ của tín hiệu OFDM” mà phải tham chiếu đến nhãn đúng của hình bằng lệnh `\ref{fig:nhãn-hình}`. Trong báo cáo mẫu này nhãn của hình phổ OFDM được đặt là “fig:pho-ofdm” và ta có thể tham chiếu đến Hình 2.16 bằng lệnh `\ref{fig:pho-ofdm}`.

Ngoài ra, khi tham khảo trong báo cáo mẫu sinh viên sẽ thấy số 0.75 trong phần lệnh vẽ hình. Con số này nghĩa là scale hình lại bằng 0.75 độ rộng văn bản. Sinh viên có thể chỉnh số này để có hình nhỏ hoặc lớn hơn sao cho có thể xem được rõ.

Trong quá trình chỉnh sửa, khi thay đổi vị trí hình, danh sách hình sẽ được cập nhật tự động.

2.3.5 Cách thêm và sử dụng tài liệu tham khảo

Khi trình bày một nội dung, phương pháp, kết quả phân tích đã ghi trong một tài liệu khác, hoặc mô hình toán chứng minh trong một tài liệu khác, sinh viên bắt buộc phải



Hình 2.16: Phổ tín hiệu OFDM

dẫn nguồn tài liệu tham khảo. Trong đó, quan trọng nhất đối với luận văn đại học là tham khảo cho công thức sử dụng trong báo cáo. Đối với các công thức kinh điển như hệ phương trình Maxwell hay giới hạn Shannon, sinh viên có thể ghi trực tiếp không cần tham khảo. Đối với các công thức khác, cần ghi rõ nguồn tham khảo.

Ví dụ 2.3.2 Công thức tính số bit tối ưu cho mỗi sóng mang trong một hệ thống điều chế đa sóng mang là [1]:

$$b_n = \log_2(1 + \gamma SNR_n). \quad (2.4)$$

Trong đó, b_n (bit) là số bit cho sóng mang thứ n , γ là hệ số tối ưu không đơn vị, và SNR_n là thông số SNR cho sóng mang thứ n .

Mỗi tài liệu tham khảo trong danh sách phải đều phải được tham chiếu đến ít nhất một điểm trong nội dung. Không được viết các tài liệu tham khảo không sử dụng trong nội dung.

Ngoài ra, đối với sinh viên Điện tử - Viễn thông, tài liệu tham khảo phải viết theo chuẩn của Hội Kỹ sư Điện và Điện tử, Institute of Electrical and Electronics Engineers (IEEE). Có rất nhiều sinh viên bị mất điểm khi bảo vệ luận văn trước hội đồng vì cách trình bày tài liệu tham khảo sai.

Trong mẫu báo cáo này, tất cả yêu cầu trình bày tài liệu tham khảo nói trên đã được tự động và chuẩn hóa theo IEEE. Syntax sử dụng theo dạng BibTex của Latex. Khi sinh viên cần lấy trích dẫn từ một nguồn chính thống, chẳng hạn trang web của IEEE (<https://ieeexplore.ieee.org/Xplore/home.jsp>), sinh viên cần làm các bước sau:

- Tìm bài báo trên trang IEEE, chẳng hạn bài báo sau: <https://ieeexplore.ieee.org/document/380108>,
- Nhấn nút “Cite this” để trích thông tin tham khảo,
- Nhấn nút “BibTex” để copy thông tin tham khảo định dạng BibTex,
- Copy thông tin trên vào file “tailieuthamkhao.bib” trong thư mục “Reference”.

Khi copy thông tin trên, sinh viên sẽ thấy dòng đầu tiên có dạng “@ARTICLE{380108,”. Chữ “ARTICLE” nghĩa là đây là một bài báo tạp chí khoa học. Dãy số “380108” chính là nhãn dùng để tham chiếu đến tài liệu này. Để thuận tiện, sinh viên nên sửa lại nhãn này cho có ý nghĩa hơn, chẳng hạn sửa lại thành “@ARTICLE{bai-bao-ve-bit-loading,”. Khi viết đến chỗ cần tham khảo tài liệu trong đoạn văn, sinh viên dùng lệnh `\cite{nhãn-tài-liệu}`. Chẳng hạn, `\cite{bai-bao-ve-bit-loading}` sẽ tạo ra tham chiếu đến tài liệu [1].

Ở mức độ luận văn tốt nghiệp đại học, nhiều sinh viên tham khảo đến các nguồn khác như sách, luận văn khóa trước, và trang web. Các nguồn này thông thường không có thông tin định dạng BibTex tự động mà cần điền một cách thủ công. Sinh viên có thể mở file “tailieuthamkhao.bib” bằng một chương trình soạn văn bản bất kỳ để xem các ví dụ cách điền tài liệu tham khảo dạng sách [2], luận văn [3], và trang web [4].

2.4 Kết luận chương

Phần này đưa ra tổng kết chung về chương. Nếu luận văn hướng đến việc ứng dụng và so sánh các phương pháp, cần nêu điểm mạnh và điểm yếu của từng phương pháp. Từ đó, giải thích được tại sao luận văn chọn phương pháp A hoặc B để trả lời câu hỏi nghiên cứu.

Ví dụ 2.4.1 Trong chương này, cơ sở lý thuyết về VLC, OFDM, và CNN đã được tổng kết và giới thiệu. Các lý thuyết này được ứng dụng phương pháp nghiên cứu được trình bày trong chương tiếp theo.

Chương 3. KẾT QUẢ NGHIÊN CỨU

3.1 Phương pháp tiếp cận

Lưu ý chương này có tỉ lệ rất lớn trong điểm số của luận văn, nên sinh viên cần hết sức lưu ý việc trình bày nội dung cũng như phân tích các kết quả.

Nội dung của mục này là miêu tả về phương pháp sinh viên đã sử dụng để có các kết quả trong báo cáo. Nếu sinh viên sử dụng phương pháp mô phỏng bằng phần mềm, thì trong mục này cần có các thông tin sau:

- Lưu đồ giải thuật,
- Giải thích các khối trong lưu đồ giải thuật,
- Các thông số điều kiện đầu vào của mô phỏng.

Nếu sinh viên thiết kế phần cứng, thì trong mục này cần có các thông tin sau:

- Sơ đồ khối hệ thống,
- Giải thích các khối trong sơ đồ,
- Sơ đồ mạch của các khối với đầy đủ thông tin về thông số linh kiện (mã số và giá trị).

Nếu sinh viên đo đạc thực nghiệm, thì trong mục này cần có các thông tin sau:

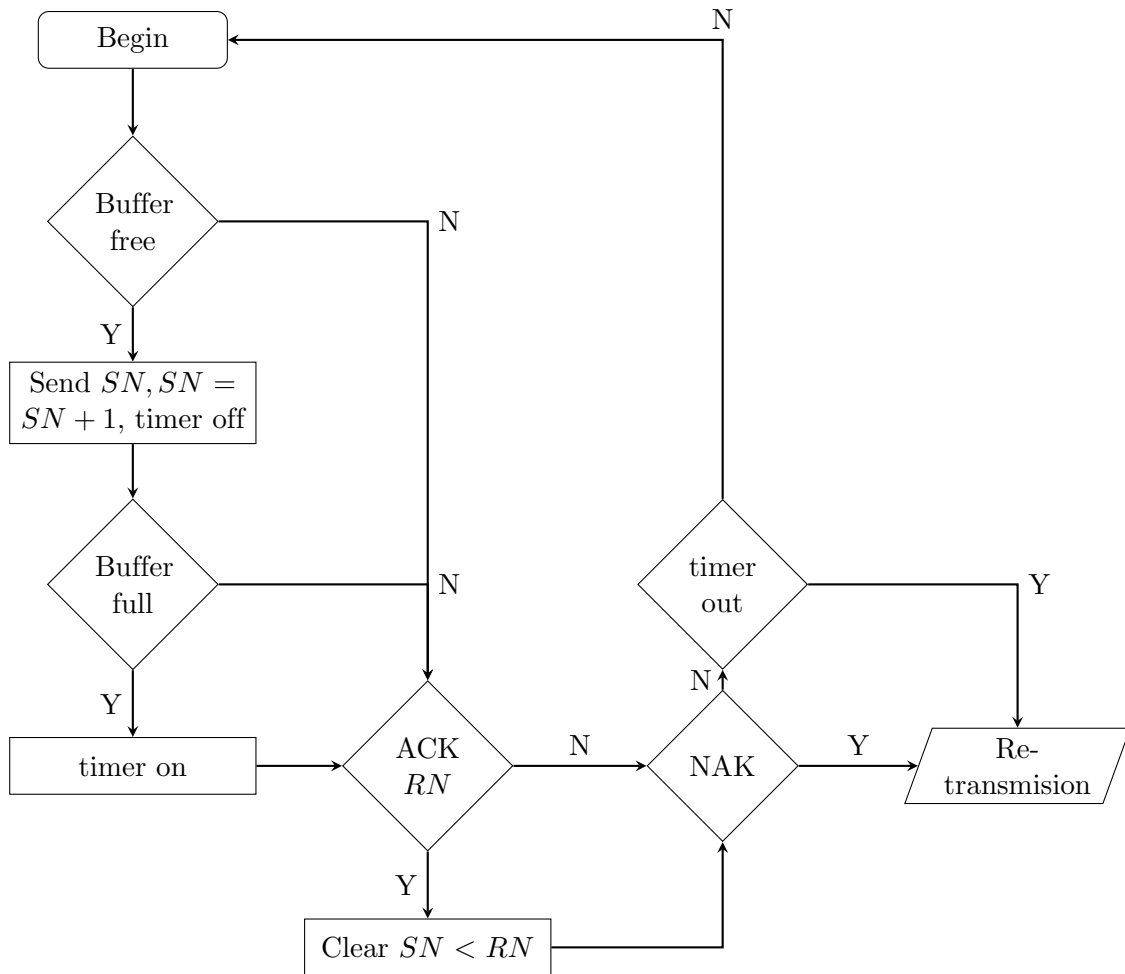
- Sơ đồ khối hệ thống thí nghiệm,
- Giải thích các khối trong sơ đồ,
- Mô tả các bước thực hiện thí nghiệm và các điều kiện khi thực hiện (thông số máy phát, máy thu, và điều kiện thí nghiệm như khoảng cách, công suất v.v).

Ví dụ 3.1.1 *Sinh viên thiết kế một hệ thống phần cứng để giao tiếp với các cảm biến, có phần mềm cho vi điều khiển, và có thí nghiệm đo đạc kết quả. Mục này chia thành các đề mục nhỏ để mô tả riêng cho các nội dung tương ứng như sau:*

- *Hệ thống cảm biến*
- *Giải thuật điều khiển và thu thập dữ liệu*
- *Điều kiện thí nghiệm*

Trong trường hợp sinh viên tự chứng minh được một công thức mới, hoặc tự đề xuất được một giải thuật mới chưa trình bày trong Chương 2, quá trình chứng minh hoặc giải thuật mới sẽ được trình bày chi tiết trong mục này.

Việc vẽ sơ đồ khối và lưu đồ giải thuật có thể được thực hiện bằng thư viện “tikzpicture” đã khai báo sẵn trong mẫu báo cáo này. Một ví dụ về giải thuật cho một giao thức điều khiển lỗi và điều khiển luồng trong môn học Truyền số liệu và Mạng thể hiện trong Hình 3.1. Một ví dụ khác về sơ đồ khối thí nghiệm truyền tín hiệu bằng hệ thống VLC thể hiện trong Hình 3.2. Sinh viên có thể dùng các chương trình khác để tạo sơ đồ khối nếu muốn.

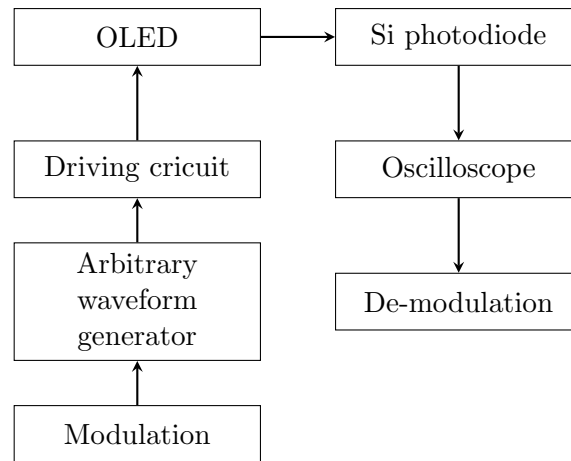


Hình 3.1: Lưu đồ giải thuật phía phát của giao thức Go back N

3.2 Kết quả và phân tích

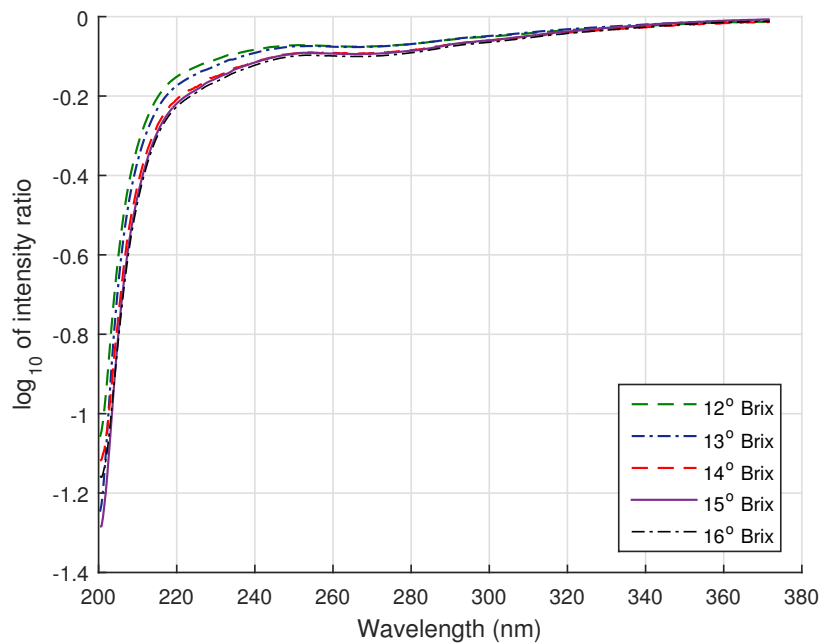
Trong mục này, sinh viên trình bày các kết quả mô phỏng và đo đạc dưới dạng hình vẽ và bảng số liệu.

Đối với hình vẽ, sinh viên không được copy màn hình hoặc vẽ bằng chương trình Excel. Sinh viên cần lưu các đồ thị kết quả ở dạng file “.fig” của Matlab. Sau đó, dùng lệnh “save as” trong cửa sổ đồ thị của Matlab để lưu hình ảnh thành dạng file “.eps”. Sinh viên copy file “.eps” vào thư mục “Figure” và dùng cách chèn hình ảnh như trong Chương 2. Lý do của các bước này là file “.eps” là hình ảnh dạng vector, nên bảo đảm không bị nhòe hoặc bể hình khi chèn vào văn bản. Một ví dụ hình vẽ dạng này thể hiện trong Hình 3.3. Sinh



Hình 3.2: Sơ đồ thí nghiệm hệ thống VLC

viên có thể so sánh Hình 3.3 với Hình 2.16 sẽ thấy sự khác biệt trong chất lượng hình ảnh. Hơn nữa, việc lưu đồ thị dạng “.fig” giúp cho việc chỉnh sửa thông tin trên đồ thị, thêm thông số và trích thông số rất nhanh và tiện lợi.



Hình 3.3: Đặc trưng nước đường từ 12 °Brix đến 16 °Brix

Khi tham khảo Hình 3.3, sinh viên cần lưu ý các yêu cầu cơ bản của một đồ thị là phải các đủ tên và đơn vị của các trục. Nếu có nhiều đồ thị trong một hình vẽ, sinh viên cần có bảng “legend” mô tả các đồ thị. Các yêu cầu này có thể được thực hiện chính xác và nhanh chóng khi vẽ bằng Matlab.

Việc tạo bảng bằng code của Latex khá rườm rà. Sinh viên có thể tạo bảng dễ dàng hơn bằng cách sử dụng một chương trình bảng tính khác (chẳng hạn như Excel) để tạo bảng, sau đó dùng các trang web chuyển đổi tự động sang dạng code của Latex. Một ví dụ của bảng dùng Latex thể hiện trong Bảng 3.1.

Lưu ý hình vẽ và bảng biểu bắt buộc phải đi kèm phân tích. Các yếu tố có thể phân tích là ý nghĩa các cực trị, nguyên nhân tăng giảm, nguyên nhân khác biệt giữa các kết quả.

Bảng 3.1: Kết quả xác suất phân loại nhãn bằng CNN

Nhãn	Kết quả phân loại nhãn					
	A	B	C	D	E	F
A	0.998	0	0.002	0	0	0
B	0	0.998	0	0	0	0.002
C	0	0	0.998	0.002	0	0
D	0	0	0.003	0.996	0.001	0
E	0	0	0	0.001	0.999	0
F	0	0.003	0	0	0	0.997

3.3 Kết luận chương

Kết luận ngắn chương 3. Tóm tắt lại các ý phân tích và trả lời các câu hỏi nghiên cứu.

Chương 4. KẾT LUẬN

4.1 Tóm tắt và kết luận chung

Trong phần kết luận, trả lời các câu hỏi nghiên cứu đã đặt ra ở chương 1 dựa vào kết quả thực hiện và phân tích trong chương 3. Nêu các điểm đạt, chưa đạt của luận văn. Nguyên nhân đạt, nguyên nhân chưa đạt và cách khắc phục nếu có.

4.2 Hướng phát triển

Phần hướng phát triển nêu ngắn gọn, không tràn lan.

4.3 Mục tiêu và tiến độ thực hiện

Nếu sinh viên dùng mẫu này để báo cáo đề cương luận văn, thay vì mục “Hướng phát triển” sinh viên cần mục này. Trong mục này, sinh viên nêu các mục tiêu dự định thực hiện trong luận văn. Kế hoạch thực hiện cần vẽ dưới dạng bảng biểu.

Ví dụ 4.3.1 *Mục tiêu dự kiến thực hiện khi hoàn thành luận văn tốt nghiệp bao gồm:*

1. Hoàn chỉnh chương trình mô phỏng tín hiệu OFDM.
2. Thực hiện mô phỏng hệ thống trong mô hình kênh truyền fading chậm.
3. Ước lượng và phân tích giá trị SNR của hệ thống.

Kế hoạch thực hiện như Bảng 4.1:

Bảng 4.1: Tiến độ thực hiện luận văn

Nội dung	Tháng			
	1	2	3	4
Tìm hiểu lý thuyết	○			
Viết chương trình	○	○		
Chạy mô phỏng		○	○	
Phân tích kết quả			○	○
Viết báo cáo				○

TÀI LIỆU THAM KHẢO

- [1] P. S. Chow, J. M. Cioffi, and J. A. C. Bingham, “A practical discrete multitone transceiver loading algorithm for data transmission over spectrally shaped channels,” *IEEE Transactions on Communications*, vol. 43, no. 2/3/4, pp. 773–775, Feb 1995.
- [2] B. D. Cullity, *Introduction to Magnetic Materials*. Reading, MA: Addison-Wesley, 1972.
- [3] A. Karnik, “Performance of TCP congestion control with rate feedback,” M. Eng. thesis, Indian Institute of Science, Bangalore, India, Jan. 1999.
- [4] (2015) The IEEE website. [Online]. Available: <http://www.ieee.org/>

Phụ lục A. CODE CHƯƠNG TRÌNH

Nếu báo cáo không có phụ lục, sinh viên tìm và xóa dòng `\input{text/phuluc.tex}` trong file “thesis.tex”. Code chương trình có thể để vào phụ lục A và các chứng minh toán học hỗ trợ có thể để vào phụ lục B. Việc lấy code và định dạng được Latex thực hiện tự động.

Sinh viên tham khảo file “phuluc.tex” để biết cách đưa code vào báo cáo. Trong lệnh `\lstinputlisting`, báo cáo mẫu có đường dẫn đến file “spectrum.m” trong thư mục con “Code” nằm chung với thư mục chứa báo cáo. Sinh viên có thể copy các file code, chẳng hạn code Matlab, vào thư mục “Code” và tham khảo file “phuluc.tex” để biết cách đưa code vào báo cáo. Để tiện lợi hơn, sinh viên có thể chỉnh đường dẫn trong lệnh này đến thư mục chứa code của mình để Latex có thể cập nhật trực tiếp. Để định dạng theo các ngôn ngữ lập trình khác nhau, sinh viên thay giá trị “Matlab” trong lệnh thành các giá trị khác phù hợp như “C++” hay “Python”.

Sau đây là ví dụ code Matlab:

```
1 fs = 100; % sample frequency (Hz)
2 t = 0:1/fs:10-1/fs; % 10 second span time vector
3 x = (1.3)*sin(2*pi*15*t) ... % 15 Hz component
4 + (1.7)*sin(2*pi*40*(t-2)) ... % 40 Hz component
5 + 2.5*gallery('normaldata',size(t),4); % Gaussian noise;
6 y = fft(x);
7 n = length(x); % number of samples
8 f = (0:n-1)*(fs/n); % frequency range
9 power = abs(y).^2/n; % power of the DFT
10
11 plot(f,power)
12 xlabel('Frequency')
13 ylabel('Power')
```