

GENETIC CNN

University of Information Technology – Vietnam National University Ho Chi Minh City

December 27, 2020

Giảng viên:
PGS.TS. LÊ ĐÌNH DUY

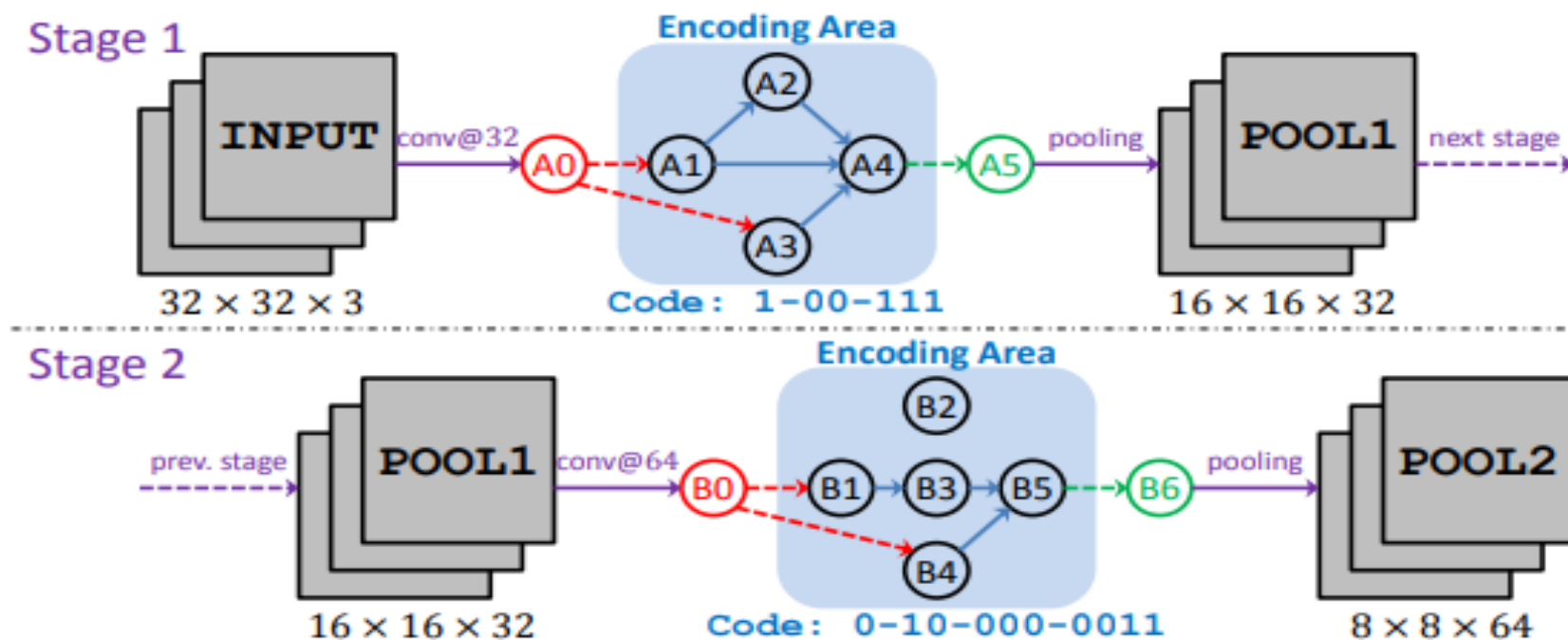
Thành viên:
1. Trịnh Hoài Anh Tú – 16521840
2. Hoàng Minh Anh – 16520035

GENETIC CNN

Nội dung:

1. Mã hóa kiến trúc mạng CNN thành chuỗi nhị phân.
2. Áp dụng giải thuật di truyền.
3. Kết quả thực nghiệm và đánh giá.

1. Mã hóa kiến trúc CNN thành chuỗi nhị phân



Đây là một kiến trúc mạng gồm 2 stage ($S=2, (K1, K2)=(4, 5)$) và chuỗi nhị phân thể hiện sự kết nối giữa các node.

1. Mã hóa kiến trúc CNN thành chuỗi nhị phân

- Một kiến trúc mạng bao gồm S đoạn (stage), và tại đoạn s -th ($s=1,2,3\dots S$) chứa K_s node, ký hiệu là V_s, k_s , $k_s = 1, 2, \dots, K_s$.
- Một stage tương đương với một khối bao gồm các layers Convolution và Pooling. Một node tương đương với một Convolution layer.
- Ở mỗi stage, chúng ta sử dụng $1 + 2 + \dots + (K_s - 1) = \frac{1}{2}K_s (K_s - 1)$ bits để mã hóa sự kết nối giữa các nodes.

1. Mã hóa kiến trúc CNN thành chuỗi nhị phân

- Bit đầu tiên biểu diễn sự kết nối giữa $(V_{s,1}; V_{s,2})$, các bit tiếp theo biểu diễn cho sự kết nối giữa $(V_{s,1}; V_{s,3})$, $(V_{s,2}; V_{s,3})$,...

Với một kiến trúc mạng gồm S đoạn và mỗi đoạn chứa K_s node thì sẽ có một chuỗi nhị phân có độ dài L để biểu diễn sự kết nối.

$$L = \frac{1}{2} \sum_s K_s (K_s - 1)$$

1. Mã hóa kiến trúc CNN thành chuỗi nhị phân

- Để đảm bảo các chuỗi nhị phân có giá trị, xác định hai node mặc định ở từng stage
- Node mặc định nhận input, ký hiệu là $V_{s,0}$, nhận dữ liệu từ stage trước, thực hiện quá trình convolution và truyền tải output của nó đến những node không nhận data từ các node khác trong vùng Encoding Area, ví dụ như node $V_{s,1}$
- Node mặc định nhận output, ký hiệu là V_{s,K_s+1} , nhận data từ tất cả các node không có truyền tải data của node đó cho các node trong vùng Encoding Area, thực hiện quá trình tổng hợp, convolution và truyền tải output của nó đến Pooling layer
- Không mã hóa sự kết nối giữa các Node trong vùng Encoding Area với các Node mặc định.

2. Áp dụng giải thuật di truyền

Thuật giải di truyền là một metaheuristic được lấy cảm hứng từ sự chọn lọc tự nhiên của thuyết tiến hóa

Thuật giải di truyền nhằm mục đích tìm ra giải pháp tối ưu, đối với đề tài này, mục đích áp dụng thuật giải di truyền là để tìm ra được mạng tích chập CNN tối ưu về hiệu suất và độ chính xác.

2. Áp dụng giải thuật di truyền

Khởi tạo:

Khởi tạo ngẫu nhiên một tập các kiến trúc mạng CNN: $\{M_{0,n}\}_{n=1}^N$ hay có thể là một quần thể

Một kiến trúc mạng được mã hóa bằng L bit. Mỗi bit được phân phối theo xác suất Bernoulli.

Sau đó chúng ta tiến hành lượng giá các cá thể để có được giá trị của hàm thích nghi (giá trị quyết định khả năng tồn tại ở các thế hệ tiếp theo).

2. Áp dụng giải thuật di truyền

Selection (Chọn lọc):

Quá trình chọn lọc sẽ được tiến hành ở đầu mỗi thế hệ.

Trước thế hệ thứ t của cá thể n , ký hiệu là $\bar{M}_{t-1,n}$ nhận được một giá trị hàm thích nghi (fitness function) $r_{t-1,n}$, được xác định là tỉ lệ nhận diện (recognition rate) mà cá thể đó nhận được ở thế hệ trước hoặc trong lúc khởi tạo. Đây là giá trị sẽ quyết định khả năng tồn tại của cá thể trong quá trình chọn lọc.

Sử dụng quá trình Russia roulette để quyết định cá thể nào có khả năng tồn tại.

2. Áp dụng giải thuật di truyền

Selection (Chọn lọc):

Từng cá thể thuộc thế hệ kế tiếp $M_{t,n}$ được xác định bởi một cá thể nào đó thuộc quần thể của thế hệ trước $\{M_{t-1,n}\}_{n=1}^N$.

Xác suất của cá thể $M_{t-1,n}$ có tỉ lệ $f_{t-1,n} = f_{t-1,0}$, trong đó $f_{t-1,0} = \min_{n=1}^N \{f_{t-1,n}\}$ là giá trị hàm thích nghi tối thiểu của thế hệ trước.

Những cá thể tốt sẽ được chọn lọc và những cá thể xấu sẽ bị loại bỏ.

2. Áp dụng giải thuật di truyền

Mutation (đột biến) và Lai ghép (Crossover):

Quá trình Mutation diễn ra trên một cá thể $M_{t,n}$ nào đó bằng cách thay đổi các bit với xác suất qM , ví dụ như 0.05.

Quá trình Crossover diễn ra bằng cách thay đổi các stage giữa các cá thể với một xác suất qC .

Xác suất Đột biến và Lai ghép của mỗi cá thể ký hiệu lần lượt là pM và pC .

2. Áp dụng giải thuật di truyền

Evaluation (Lượng giá):

Mỗi cá thể sẽ nhận một giá trị hàm thích nghi bằng cách lượng giá.

Sử dụng một dataset. Train từng cá thể lại từ đầu thông qua dataset đó.

Nếu một cá thể của thế hệ t nào đó đã được lượng giá, chúng ta cũng lượng giá lại một lần nữa và tính ra độ chính xác trung bình

2. Áp dụng giải thuật di truyền

Flowchart thuật toán:

Input: Một tập dữ liệu D, số thế hệ T, số các cá thể trong quần thể N, xác suất của quá trình đột biến và lai ghép lần lượt là pM và pC, các tham số đột biến và lai ghép qM và qC.

Khởi tạo: Khởi tạo ngẫu nhiên một tập các cá thể $\{M_{0,n}\}_{n=1}^N$, và tính toán độ chính xác của từng cá thể

For t=1,2,3...T

Selection: Tạo ra một thế hệ mới bằng cách sử dụng Russian Roulette trên tập thế hệ cũ.

Crossover: Với mỗi cặp cá thể $\{(M_{t,2n-1}, M_{t,2n})\}_{n=1}^{\lfloor N/2 \rfloor}$ thực hiện quá trình lai ghép với xác suất là pC và tham số qC.

Mutation: Với các cá thể không lai ghép $\{M_{t,n}\}_{n=1}^N$, thực hiện quá trình đột biến với xác suất pM và tham số qM.

Evaluation: Tính toán độ chính xác với từng cá thể mới $\{M_{t,n}\}_{n=1}^N$

End for

Output: Một tập các cá thể $\{M_{T,n}\}_{n=1}^N$ với độ chính xác.

3. Kết quả thực nghiệm và đánh giá

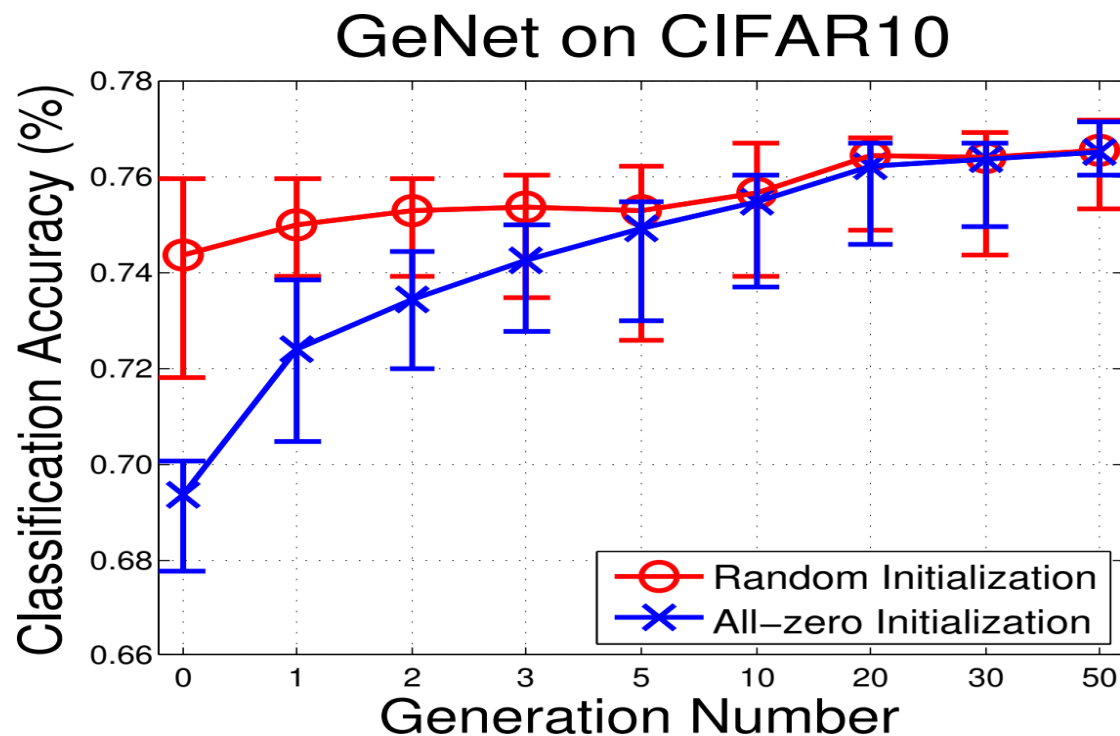
Thử nghiệm CIFAR10

1. Tập dữ liệu (Dataset) CIFAR10
2. Cá thể ban đầu (N = 20 cá thể)
3. Quá trình di truyền (Genetic) (T = 50 vòng)
4. Các thông số khác ($p_M = 0.8$, $q_M = 0.05$, $p_C = 0.2$, $q_C = 0.2$).

Gen	Max %	Min %	Avg %	Med %	Std-D	Best Network Structure
00	75.96	71.81	74.39	74.53	0.91	0-01 0-01-111 0-11-010-0111
01	75.96	73.93	75.01	75.17	0.57	0-01 0-01-111 0-11-010-0111
02	75.96	73.95	75.32	75.48	0.57	0-01 0-01-111 0-11-010-0111
03	76.06	73.47	75.37	75.62	0.70	1-01 0-01-111 0-11-010-0111
05	76.24	72.60	75.32	75.65	0.89	1-01 0-01-111 0-11-010-0011
08	76.59	74.75	75.77	75.86	0.53	1-01 0-01-111 0-11-010-1011
10	76.72	73.92	75.68	75.80	0.88	1-01 0-01-110 0-11-111-0001
20	76.83	74.91	76.45	76.79	0.61	1-01 1-01-110 0-11-111-0001
30	76.95	74.38	76.42	76.53	0.46	1-01 0-01-100 0-11-111-0001
50	77.19	75.34	76.58	76.81	0.55	1-01 0-01-100 0-11-101-0001

3. Kết quả thực nghiệm và đánh giá

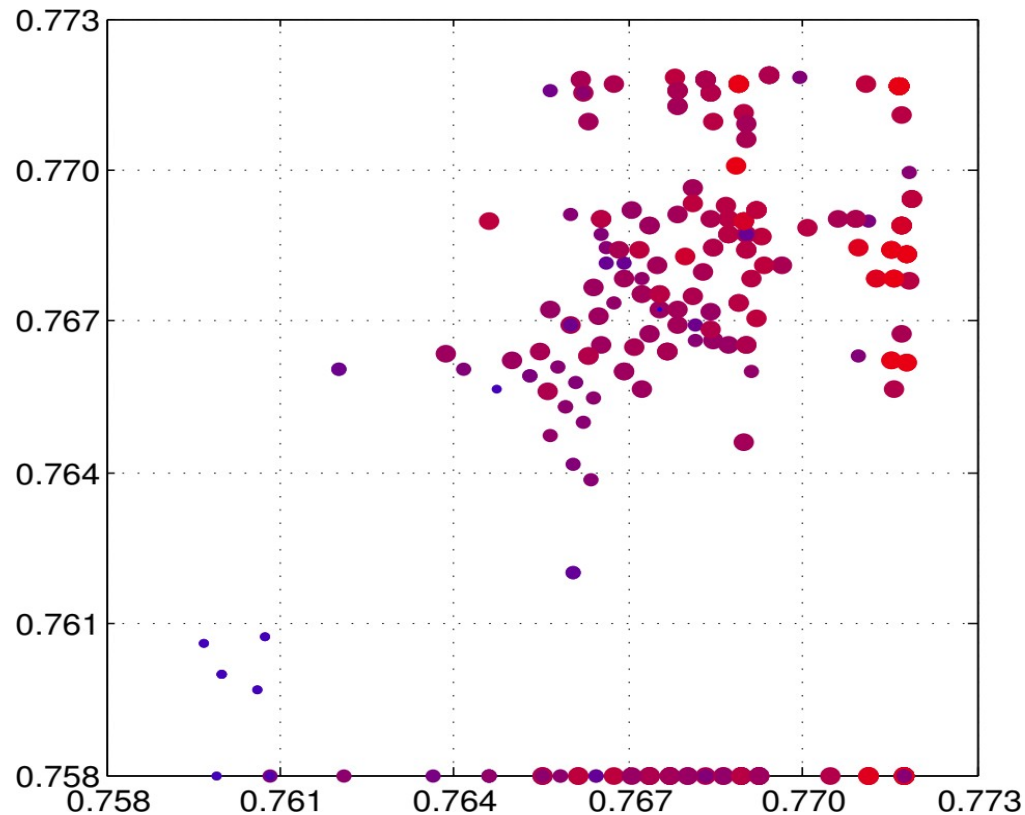
Cá thể Naive và cá thể Non-naive



Độ chính xác nhận dạng trung bình trên tất cả các cá thể liên quan đến số thế hệ.

3. Kết quả thực nghiệm và đánh giá

Tính hợp lý và độ hiệu quả



Mối quan hệ trong độ chính xác giữa parent(s) và child(ren).

3. Kết quả thực nghiệm và đánh giá

Thí nghiệm Transfer quy mô nhỏ

	SVHN	CF10	CF100
Zeiler <i>et.al</i> [43]	2.80	15.13	42.51
Goodfellow <i>et.al</i> [10]	2.47	9.38	38.57
Lin <i>et.al</i> [26]	2.35	8.81	35.68
Lee <i>et.al</i> [24]	1.92	7.97	34.57
Liang <i>et.al</i> [25]	1.77	7.09	31.75
Lee <i>et.al</i> [23]	1.69	6.05	32.37
Zagoruyko <i>et.al</i> [42]	1.77	5.54	25.52
Xie <i>et.al</i> [39]	1.67	5.31	25.01
Huang <i>et.al</i> [16]	1.75	5.25	24.98
Huang <i>et.al</i> [15]	1.59	3.74	19.25
GeNet after G-00	2.25	8.18	31.46
GeNet after G-05	2.15	7.67	30.17
GeNet after G-20	2.05	7.36	29.63
GeNet #1 (G-50)	1.99	7.19	29.03
GeNet #2 (G-50)	1.97	7.10	29.05
GeNet from WRN [42]	1.71	5.39	25.12

So sánh tỷ lệ lỗi nhận dạng (%) với trạng thái hiện đại (State of the arts).

3. Kết quả thực nghiệm và đánh giá

Thí nghiệm Transfer quy mô lớn

	Top-1	Top-5	# Paras
AlexNet [19]	42.6	19.6	62M
GoogLeNet [36]	34.2	12.9	13M
VGGNet-16 [32]	28.5	9.9	138M
VGGNet-19 [32]	28.7	9.9	144M
GeNet #1	28.12	9.95	156M
GeNet #2	27.87	9.74	156M

Tỷ lệ lỗi nhận dạng Top-1 và Top-5 (%) trên bộ dữ liệu ILSVRC2012.

3. Kết quả thực nghiệm và đánh giá

Tiến hành so sánh chúng với các kiến trúc CNN truyền thống khác nhau trên cùng dataset *ILSVRC2012* [3] (số liệu lấy từ trang [4]) và ta thu được kết quả như bảng:

	Top-1	Top-5	# Paras
AlexNet [19]	42.6	19.6	62M
GoogLeNet [36]	34.2	12.9	13M
VGGNet-16 [32]	28.5	9.9	138M
VGGNet-19 [32]	28.7	9.9	144M
GeNet #1	28.12	9.95	156M
GeNet #2	27.87	9.74	156M

Dựa vào kết quả trên ta có thể đánh giá được hai kiến trúc mạng phát sinh có kết quả Top-1 và Top-5 error tốt hơn những kiến trúc truyền thống ban đầu

Tham khảo

- [1] *Lingxi Xie, Alan L. Yuille: **Genetic CNN**. ICCV 2017: 1388-1397*
- [2] *Dataset: CIFAR10*
(<http://www.cs.toronto.edu/~kriz/cifar.html>).
- [3] *Dataset: ILSVRC2012*
(<https://dbcollection.readthedocs.io/en/latest/datasets/imagenet.html>)
- [4] <https://www.vlfeat.org/matconvnet/pretrained/>
- [5] *L. Xie, J. Wang, Z. Wei, M. Wang, and Q. Tian. DisturbLabel: **Regularizing CNN on the Loss Layer**. CVPR 2016: 4753-4762*