

# TÁC ĐỘNG CỦA DATA AUGMENTATION TRÊN CIFAR-10

## 1. Giới thiệu và Mục tiêu

Nhằm đánh giá một cách định lượng tác động của kỹ thuật tăng cường dữ liệu (Data Augmentation) lên hiệu suất và tốc độ hội tụ của mô hình phân loại ảnh. Tập trung vào các mục tiêu sau:

1. Chọn lọc 5 lớp từ tập dữ liệu CIFAR-10, mỗi lớp bao gồm 1000 ảnh, tạo thành một tập dữ liệu con 5000 ảnh cho huấn luyện và 5000 ảnh cho kiểm tra.
2. Áp dụng các kỹ thuật Data Augmentation đa dạng cho tập huấn luyện.
3. Huấn luyện một mô hình mạng nơ-ron tích chập (CNN) với hai cấu hình: có và không có Data Augmentation.
4. Thực hiện lặp lại quá trình huấn luyện 3 lần cho mỗi cấu hình để thu thập kết quả trung bình và đánh giá độ ổn định.
5. So sánh chi tiết hiệu năng (độ chính xác, loss) và tốc độ hội tụ (thời gian huấn luyện) giữa hai cấu hình.
6. Sử dụng nền tảng WandB (Weights & Biases) để theo dõi, ghi nhật ký và trực quan hóa toàn bộ quá trình thử nghiệm.
7. Xây dựng một giao diện dự đoán đơn giản bằng Gradio để minh họa khả năng của mô hình tốt nhất.

## 2. Phương pháp

### 2.1. Chuẩn bị Dữ liệu

- **Tập dữ liệu gốc:** CIFAR-10.
- **Các lớp được chọn:** 5 lớp bao gồm 'airplane', 'automobile', 'bird', 'cat', 'deer'.
- **Số lượng mẫu:** 1000 mẫu cho mỗi lớp trong tập huấn luyện (tổng 5000 mẫu). Tập kiểm tra bao gồm tất cả các mẫu thuộc 5 lớp này (tổng 5000 mẫu).
- **Chuẩn hóa:** Tất cả ảnh đều được chuẩn hóa với giá trị trung bình và độ lệch chuẩn của CIFAR-10.

### 2.2. Kỹ thuật Data Augmentation

- **Không Augmentation:** Chỉ áp dụng `transforms.ToTensor()` và `transforms.Normalize()`.

- **Có Augmentation:** Ngoài các bước cơ bản, áp dụng thêm:
  - `transforms.RandomHorizontalFlip(p=0.5)`
  - `transforms.RandomRotation(15)`
  - `transforms.RandomAffine(degrees=0, translate=(0.1, 0.1))`
  - `transforms.RandomResizedCrop(32, scale=(0.8, 1.0))`
  - `transforms.ColorJitter(brightness=0.2, contrast=0.2, saturation=0.2, hue=0.1)`

### 2.3. Kiến trúc Mô hình

- **Mô hình:** SimpleCNN - một mạng CNN tùy chỉnh bao gồm 3 khối tích chập, mỗi khối có 2 lớp Conv2d, BatchNorm2d, ReLU, tiếp theo là MaxPool2d và Dropout. Phần classifier gồm 2 lớp Linear với Dropout.
- **Tổng số tham số:** 1,339,557.

### 2.4. Cấu hình Huấn luyện

- **Thiết bị:** NVIDIA Tesla T4 GPU(Colab).
- **Batch Size:** 64
- **Số lượng Epoch:** 50
- **Learning Rate:** 0.001
- **Optimizer:** Adam
- **Loss Function:** `nn.CrossEntropyLoss()`
- **Learning Rate Scheduler:** `torch.optim.lr_scheduler.CosineAnnealingLR`
- **Số lần chạy:** 3 lần cho mỗi cấu hình (có/không augmentation) để lấy kết quả trung bình.

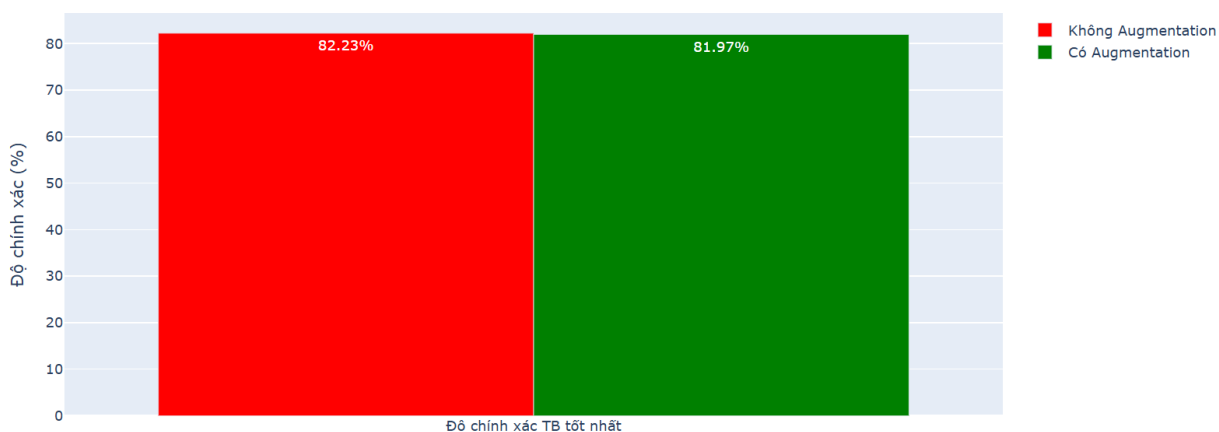
### 2.5. Đánh giá

- Theo dõi `train_loss`, `train_acc`, `test_loss`, `test_acc` qua từng epoch.
- Lưu trữ mô hình có độ chính xác test tốt nhất.
- Tính toán ma trận nhầm lẫn (Confusion Matrix) và báo cáo phân loại (Classification Report) cho mô hình tốt nhất.

### 3. Kết quả

#### 3.1. Tổng quan hiệu năng trung bình (3 lần chạy)

So sánh độ chính xác test trung bình tốt nhất



Cấu hình	Độ chính xác Test trung bình tốt nhất	Thời gian huấn luyện trung bình (s)
Không Augmentation	82.23%	148.99s
Có Augmentation	81.97%	327.98s

- **Cải thiện độ chính xác:** -0.27% (Có Augmentation thấp hơn 0.27% so với Không Augmentation).
- **Chênh lệch thời gian:** +178.99s (Có Augmentation tốn thêm 178.99s, gần gấp đôi thời gian).

#### 3.2. Phân tích chi tiết

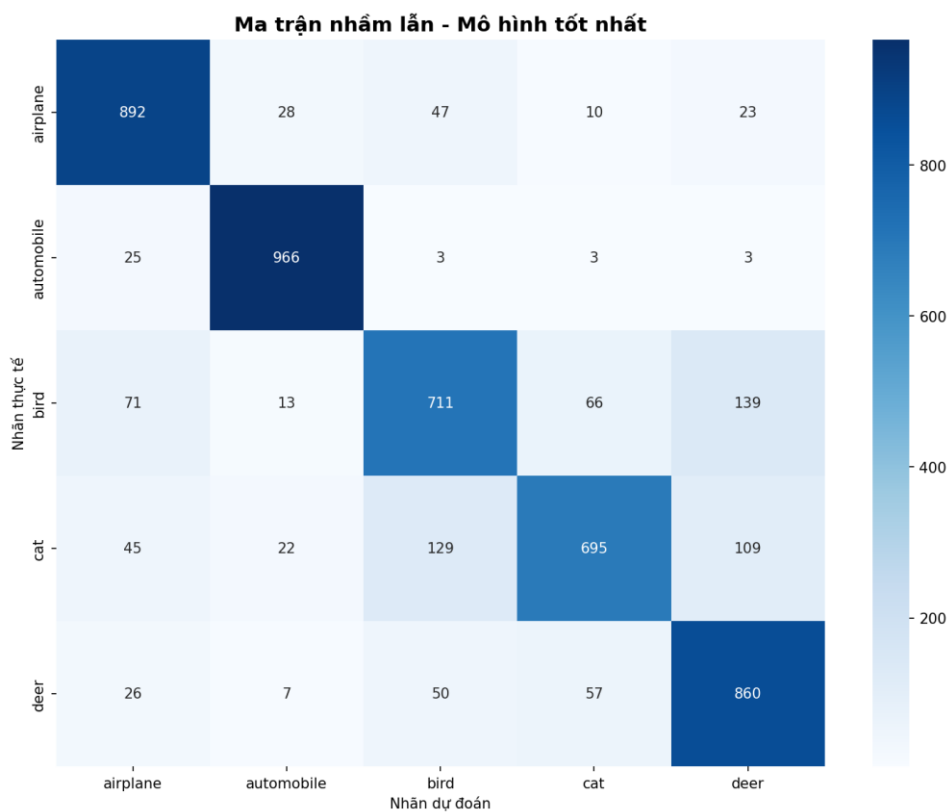
- **Độ chính xác:** Kết quả cho thấy mô hình được huấn luyện **không sử dụng augmentation** đạt độ chính xác test trung bình tốt nhất là 82.23%, cao hơn một chút so với mô hình **có augmentation** (81.97%). Điều này đi ngược lại với kỳ vọng thông thường về lợi ích của data augmentation.
- **Thời gian huấn luyện:** Mô hình **có augmentation** mất khoảng 327.98 giây để hoàn thành quá trình huấn luyện, gần gấp đôi thời gian so với mô hình **không**

**augmentation** (148.99 giây). Điều này là do các phép biến đổi ảnh được áp dụng theo thời gian thực trong mỗi epoch, làm tăng gánh nặng tính toán.

- **Độ ổn định:** Độ lệch chuẩn của độ chính xác test tốt nhất cho cấu hình không augmentation là 0.17%, trong khi của cấu hình có augmentation là 0.37%. Điều này cho thấy các lần chạy với augmentation có sự biến động kết quả lớn hơn một chút, có thể do tính ngẫu nhiên của các phép biến đổi.

### 3.3. Mô hình tốt nhất (từ cấu hình có Augmentation, lần chạy 1)

- **Độ chính xác Test:** 82.48%
- **Loss Test:** 0.4895
- **Ma trận nhầm lẫn và Báo cáo phân loại:**



- Mô hình hoạt động tốt nhất với lớp 'automobile' (precision 0.93, recall 0.97, f1-score 0.95).

- Các lớp như 'bird' (f1-score 0.73) và 'cat' (f1-score 0.76) có hiệu suất thấp hơn, đặc biệt là recall của 'cat' (0.69) cho thấy mô hình gặp khó khăn trong việc nhận diện lớp này.

#### 4. Kết luận và Khuyến nghị

**Kết luận:** Trong bối cảnh của nghiên cứu này với tập dữ liệu CIFAR-10 con và kiến trúc CNN đơn giản, Data Augmentation không những không cải thiện hiệu suất mà còn làm giảm nhẹ độ chính xác và tăng đáng kể thời gian huấn luyện. Điều này gợi ý rằng các tham số augmentation được lựa chọn có thể quá mạnh hoặc không phù hợp với đặc điểm của tập dữ liệu và mô hình, dẫn đến việc tạo ra các mẫu huấn luyện bị biến đổi quá mức và gây nhiễu cho quá trình học.

#### Nhận xét và Khuyến nghị:

1. **Điều chỉnh tham số Augmentation:** Đây là yếu tố quan trọng nhất. Cần thử nghiệm với các cường độ và loại phép biến đổi augmentation khác nhau. Ví dụ, giảm mức độ xoay, dịch chuyển, hoặc chỉnh màu, hoặc loại bỏ một số phép biến đổi cụ thể. Việc tùy chỉnh augmentation phù hợp với từng dataset là then chốt.
2. **Kiểm tra tính cân bằng của Augmentation:** Đảm bảo rằng các phép biến đổi không làm mất đi thông tin quan trọng hoặc tạo ra dữ liệu quá xa rời phân phối thực của tập dữ liệu.
3. **Tăng cường kiến trúc mô hình:** Một mô hình mạnh mẽ hơn có thể tận dụng tốt hơn sự đa dạng từ augmented data. Việc sử dụng các kiến trúc SOTA (State-Of-The-Art) có thể cho kết quả khác.
4. **Kích thước tập dữ liệu:** Mặc dù augmentation thường có lợi cho các tập dữ liệu nhỏ, với 5000 ảnh cho 5 lớp, có thể tập dữ liệu vẫn chưa đủ lớn để augmentation phát huy toàn bộ sức mạnh, hoặc cần một chiến lược augmentation khác.
5. **Quan trọng của WandB:** WandB đã chứng minh được giá trị trong việc theo dõi và so sánh hiệu quả của các lần chạy khác nhau, giúp dễ dàng nhận diện và phân tích các xu hướng trong quá trình huấn luyện.
6. **Giao diện Gradio:** Việc xây dựng giao diện dự đoán giúp trực quan hóa và kiểm tra mô hình một cách nhanh chóng, mang lại cái nhìn trực quan về khả năng của mô hình trong thực tế.

Nghiên cứu này nhấn mạnh rằng Data Augmentation không phải lúc nào cũng là một 'phép màu', và việc áp dụng nó đòi hỏi sự hiểu biết sâu sắc và thử nghiệm kỹ lưỡng để đạt được hiệu quả tối ưu.