TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

BÁO CÁO PROJECT II

Sử Dụng Các Phương Pháp Học Máy Để Nhận Diện Hành Vi Bạo Lực Trong Video

BÙI THANH TÙNG
tung.bt204931@sis.hust.edu.vn

Ngành: Khoa học dữ liệu và Trí tuệ nhân tạo

Giảng viên hướng dẫn: TS. Trần Nhật Hóa

Trường: Công nghệ Thông tin và Truyền thông

HÀ NỘI, 08/2023

Sử Dụng Các Phương Pháp Học Máy Để Nhận Diện Hành Vi Bạo Lực Trong Video

BÁO CÁO - Project II 20222

Bùi Thanh Tùng - 20204931
Giảng viên hướng dẫn: TS. Trần Nhật Hoá

ABSTRACT

Trong báo cáo nghiên cứu này, em tập trung vào nhận dạng hành động có tính chất bạo lực sử dụng Bộ dữ liệu RLVS (Real Life Violence Situations). Ba mô hình khác nhau bao gồm XGBoost, ResNet50 và InceptionV3 được thử nghiệm, trong đó mô hình InceptionV3 cho kết quả tốt nhất. So với XGBoost và ResNet50, InceptionV3 đã thể hiện khả năng học tập sâu tốt hơn và khám phá các đặc trưng phức tạp từ dữ liệu hình ảnh. Cách sử dụng RLVS Dataset và áp dụng các kỹ thuật tăng cường dữ liệu phù hợp đã chứng minh tính hiệu quả của mô hình InceptionV3 trong việc nhận dạng hành động bạo lực. Em mong rằng kết quả của dự án này sẽ cung cấp một cơ sở quan trọng để phát triển các ứng dụng thực tế như giám sát video và phân loại hành vi con người trong môi trường không gian thực.

Keywords. Action Recognition, Image Classification, Deep Learning, Machine Learning.

ACKNOWLEDGEMENT

Em xin bày tỏ lòng biết ơn sâu sắc đến Tiến sĩ Trần Nhật Hóa, người đã tận tình hướng dẫn em trong Project II lần này. Nhờ có sự hỗ trợ và chỉ dẫn từ thầy, em đã hoàn thành dự án này đúng hạn và đạt được kết quả như mong đợi.

Mục lục

1 Giới thiệu

Trong thời gian gần đây, việc phân loại và nhận dạng hành động trong các tình huống thực tế đã trở thành một chủ đề quan trọng trong lĩnh vực thị giác máy tính và học máy[1]. Nhận dạng hành động có ứng dụng rộng rãi trong nhiều lĩnh vực như giám sát an ninh và nhận dạng hành vi con người. Tuy nhiên, nhận dạng các tình huống bạo lực trong thực tế vẫn là một thách thức lớn do tính phức tạp và đa dạng các thể loại của chúng[4].

Trong báo cáo Project II lần này, em tập trung vào nhận dạng hành động sử dụng Bộ dữ liệu RLVS (Real Life Violence Situations) gồm các tình huống bạo lực trong cuộc sống thực. Bộ dữ liệu này mang tính thực tế cao và đòi hỏi các mô hình nhận dạng phải đối mặt với các trường hợp phức tạp và không thể dự đoán trước. Điều này đặt ra một thách thức đáng kể trong việc xây dựng mô hình nhận dạng hiệu quả.

Để cải thiện hiệu suất nhận dạng hành động, các kỹ thuật tăng cường dữ liệu khác nhau được thực hiện trên bộ dữ liệu RLVS. Tăng cường dữ liệu giúp mô hình học được các đặc trưng chung từ dữ liệu huấn luyện và cải thiện khả năng tổng quát hóa của mô hình[5].

Em tiến hành thử nghiệm ba mô hình khác nhau để thực hiện nhận dạng hành động trên bộ dữ liệu RLVS, bao gồm XGBoost, ResNet50 và InceptionV3. Kết quả thử nghiệm cho thấy mô hình InceptionV3 đạt được hiệu suất cao nhất và vượt trội so với hai mô hình còn lại. Mô hình InceptionV3 đã thể hiện khả năng học tập sâu tốt hơn và khám phá các đặc trưng phức tạp từ dữ liệu hình ảnh, giúp đạt được kết quả nhận dạng chính xác.

Em hy vọng kết quả này sẽ cung cấp một đóng góp quan trọng cho lĩnh vực nhận dạng hành động và học máy, đồng thời sẽ hỗ trợ trong việc phát triển các ứng dụng thực tiễn như giám sát video và phân loại hành vi con người trong môi trường không gian thực.

2 Phướng pháp giải quyết

2.1 XGBoost

XGBoost, viết tắt của *eXtreme Gradient Boosting*, là một thuật toán học máy dựa trên cơ sở của việc tăng cường gradient (Gradient Boosting). Nó được thiết kế để tối ưu hóa hiệu suất và tốc độ tính toán.

2.1.1 Nguyên lý hoạt động của XGBoost

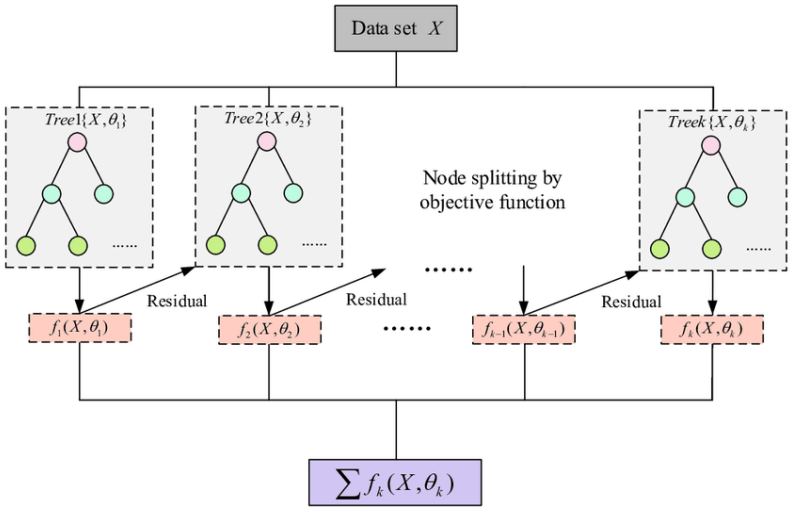
XGBoost hoạt động dựa trên nguyên tắt của mô hình tăng cường, trong đó mô hình mới được xây dựng để cải thiện lỗi của mô hình trước đó. Đặc điểm nổi bật của XGBoost so với các thuật toán tăng cường gradient truyền thống là:

- **Regularization:** XGBoost đưa ra một hình phạt regularization cho số lượng cây (leafs) và kích thước của chúng, giúp tránh được việc quá khớp (overfitting).
- **Tối ưu hóa tính toán:** Với sự cải tiến về thuật toán và khả năng song song hóa, XGBoost có thể nhanh chóng và hiệu quả xây dựng các cây quyết định.
- **Quản lý dữ liệu bị thiếu:** XGBoost có thể tự động xử lý dữ liệu bị thiếu, giảm bớt nhu cầu can thiệp từ phía người dùng.
- **Khả năng mở rộng:** XGBoost có thể mở rộng trên nền tảng dọc và ngang và có thể được tích hợp trong các khung hình phân tán như Apache Spark.

2.1.2 Cơ chế hoạt động

XGBoost[2] xây dựng mô hình theo dạng ensemble như hình 1 bằng cách kết hợp nhiều cây quyết định. Trong mỗi vòng lặp, thuật toán sẽ tạo ra một cây mới để dự đoán và giảm thiểu sai số từ mô hình hiện tại. Điều này được thực hiện thông qua việc tối ưu hóa một hàm mất mát được xác định trước, mà trong đó cả sai số dự đoán và hình phạt regularization đều được tính toán.

Quá trình này sẽ được tiếp tục cho đến khi số lượng cây đạt đến giới hạn đã được xác định trước hoặc khi mô hình không còn cải thiện đáng kể.



Hình 1: Cách mô hình XGBoost hoạt động

2.1.3 Ứng dụng trong nhận diện hành động

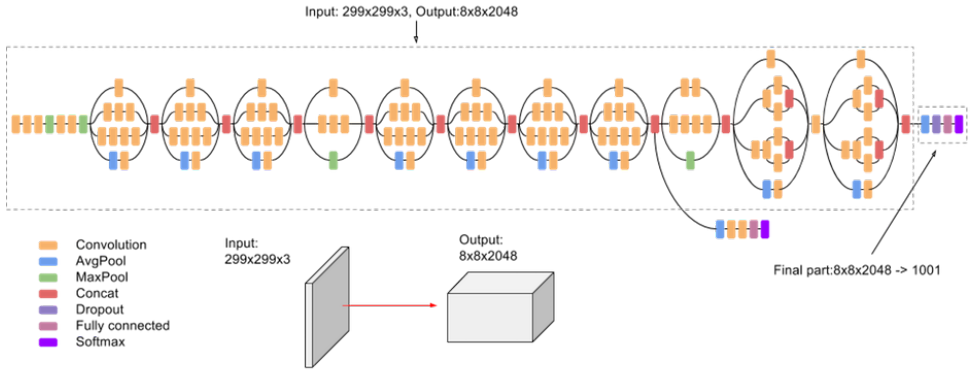
Trong bài toán nhận diện hành động, chúng ta có thể sử dụng XGBoost để phân loại các hành động dựa trên các đặc trưng được trích xuất từ dữ liệu. Mặc dù XGBoost thường được sử dụng trong các bài toán phân loại tabular data, nhưng với sự kết hợp của các đặc trưng tốt và thuật toán tối ưu, XGBoost có thể đem lại kết quả tốt trong việc nhận diện hành động.

2.2 InceptionV3

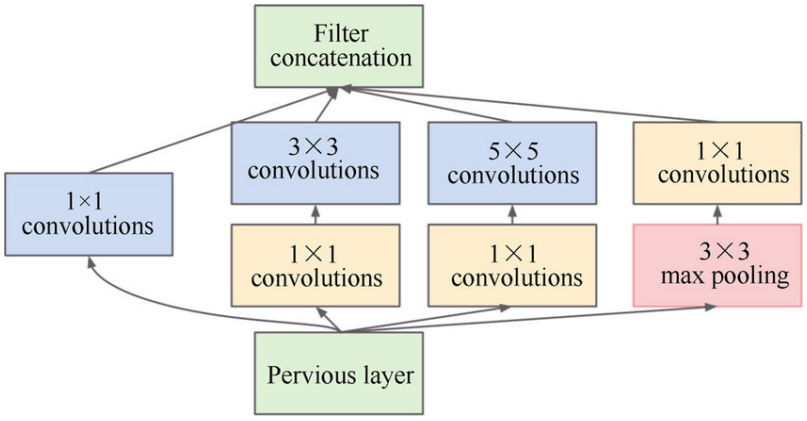
Mô hình InceptionV3[7] là một phần của dãy mô hình Inception, được thiết kế để cải thiện hiệu suất tính toán khi mở rộng độ sâu và độ rộng của mạng. Kiến trúc mô hình InceptionV3 được thể hiện trong hình 2

2.2.1 Mô-đun Inception

Trái tim của kiến trúc InceptionV3 là mô-đun Inception (được thể hiện ở hình 3). Mỗi mô-đun Inception bao gồm một tập hợp các phép tích chập với kích thước khác nhau, giúp mô hình nắm bắt được cả các thông tin cục bộ và toàn cục trong ảnh. Tất cả đầu ra từ những phép tích chập này được nối lại với nhau theo chiều sâu.



Hình 2: Kiến trúc mô hình InceptionV3



Hình 3: Mô-đun Inception

2.2.2 Factorization into Smaller Convolutions

Một trong những đổi mới quan trọng trong InceptionV3 là việc phân tích tích chập 2D lớn thành hai phép tích chập 1D liên tiếp. Ví dụ, thay vì sử dụng một phép tích chập 5×5, InceptionV3 sẽ sử dụng hai phép tích chập liên tiếp: 5×1 và 1×5. Điều này giúp giảm thiểu số lượng tham số và tăng hiệu suất tính toán mà không làm giảm độ chính xác.

2.2.3 Auxiliary Classifiers

Để giúp mô hình học được các đặc trưng tốt hơn ở những lớp sâu, InceptionV3 bổ sung các bộ phân loại phụ giữa kiến trúc. Mặc dù chúng không được sử dụng trong

quá trình dự đoán, nhưng trong quá trình huấn luyện, chúng giúp gia tăng gradient và giảm bớt vấn đề biến mất gradient.

2.2.4 Label Smoothing

Một kỹ thuật quan trọng khác được áp dụng trong InceptionV3 là làm mịn nhãn (Label Smoothing). Điều này giúp mô hình tránh được sự tự tin quá mức vào một nhãn cụ thể, giúp mô hình trở nên ổn định và chính xác hơn.

Tất cả những đổi mới trên đều được kết hợp trong mô hình InceptionV3, giúp nó có độ chính xác cao mà vẫn giữ được hiệu suất tính toán.

2.2.5 Ứng dụng trong nhận diện hành động

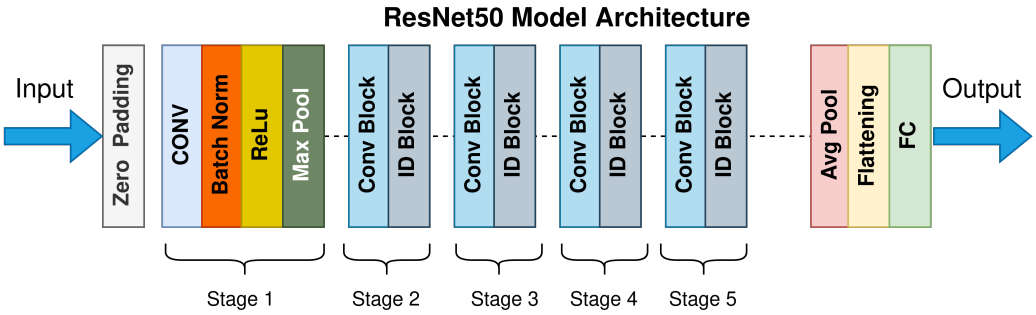
Dưới đây là cách thức bổ sung và chỉnh sửa lớp trên mô hình cơ sở InceptionV3 để thích nghi với nhiệm vụ nhận dạng hành động:

- **Lớp AveragePooling2D**: Sau khi lấy đầu ra từ mô hình cơ sở, ta thêm một lớp pooling trung bình với kích thước bể là 5×5. Mục đích của lớp này là giảm số lượng tham số và ngăn chặn việc quá khớp.
- **Lớp Flatten**: Lớp này chuyển đổi tensor đầu ra từ lớp AveragePooling2D thành một vector một chiều.
- **Lớp Dense**: Lớp này bao gồm 512 nút với hàm kích hoạt là ReLU. Hàm kích hoạt ReLU giúp mô hình học được các tính năng phi tuyến tính.
- **Lớp Dropout**: Thêm một lớp Dropout với tỷ lệ bỏ qua là 0.5 giúp ngăn chặn hiện tượng quá khớp và làm cho mô hình trở nên mạnh mẽ hơn.
- **Lớp Dense cuối cùng**: Đây là lớp đầu ra, gồm 2 nút tương ứng với 2 lớp cần phân loại. Hàm kích hoạt "softmax" được sử dụng để tính toán xác suất cho mỗi lớp.

Việc tùy chỉnh và thêm các lớp trên mô hình cơ sở InceptionV3 giúp tối ưu hóa mô hình cho nhiệm vụ nhận dạng hành động của chúng ta. Sự kết hợp giữa các lớp này giúp cải thiện độ chính xác và ngăn chặn việc mô hình quá khớp với dữ liệu trong tập huấn luyện.

2.3 Resnet50

ResNet50[3] thuộc dòng mô hình ResNet (Residual Networks) được thiết kế để giải quyết vấn đề gradient vanishing trong quá trình huấn luyện các mạng neuron sâu. Với việc sử dụng các kết nối dư (residual connections), ResNet có thể được huấn luyện đến hàng trăm, thậm chí hàng ngàn lớp mà vẫn duy trì khả năng hội tụ. Kiến trúc mô hình này được thể hiện ở hình 4



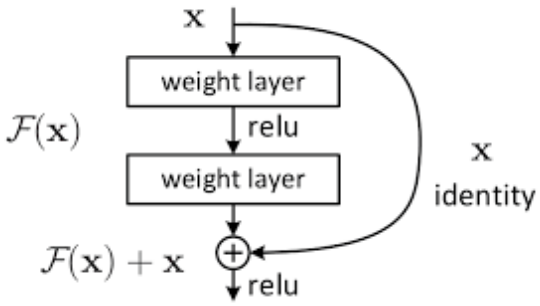
Hình 4: Kiến trúc mô hình ResNet50

2.3.1 Cơ chế hoạt động

Đặc trưng quan trọng nhất của ResNet là việc sử dụng kết nối dư (Residual Blocks). Trong một mạng truyền thống, tại mỗi lớp, đầu vào được biến đổi và truyền đến lớp tiếp theo. Tuy nhiên, trong ResNet, đầu vào được cộng trực tiếp vào đầu ra của một hoặc nhiều lớp sau (như hình 5). Điều này cho phép gradient được truyền thẳng qua các lớp mà không cần phải đi qua bất kỳ biến đổi nào, giúp giảm bớt vấn đề biến mất gradient.

Với số lượng tham số lớn, có nguy cơ mô hình sẽ quá khớp với dữ liệu huấn luyện. Tuy nhiên, kết nối dư và thiết kế bottleneck giúp giảm bớt đáng kể khả năng vấn đề này xảy ra. Trong thực tế, phần đánh giá thực nghiệm cho thấy mô hình Resnet50 làm rất tốt trong việc tổng quát hóa dữ liệu mà không bị phụ thuộc quá nhiều vào tập huấn luyện

ResNet50 có nghĩa là mô hình có 50 lớp, bao gồm cả các lớp tích chập, lớp kết nối đầy đủ và lớp pooling.



Hình 5: Residual Blocks trong ResNet50

2.3.2 Chỉnh sửa và bổ sung các lớp

Giống như phần InceptionV3, chúng ta cũng bổ sung các lớp tùy chỉnh tương tự ResNet50 để tối ưu hóa cho nhiệm vụ nhận dạng hành động. Bằng cách sử dụng kiến trúc ResNet50 kết hợp với các lớp tùy chỉnh, chúng ta có thể đạt được một mô hình hiệu quả và đạt độ chính xác cao.

2.4 Tiền xử lý

Trong phần này, cách tiền xử lý dữ liệu trước khi đưa vào huấn luyện các mô hình XGBoost, ResNet50 và InceptionV3 để thực hiện nhận dạng hành động sẽ được trình bày cụ thể và chi tiết như sau.

XGBoost: Đối với mô hình XGBoost, các bước tiền xử lý dữ liệu sau được thực hiện:

- Đọc các khung hình của mỗi video, trong đó hai khung hình liên tiếp được lấy cách nhau 15 khung hình.
- Điều chỉnh kích thước của mỗi hình ảnh thành (64, 64, 3) để giảm độ phức tạp của dữ liệu.
- Sử dụng LabelBinarizer để gán nhãn cho các hình ảnh.
- Reshape kích thước của dữ liệu sao cho mỗi hình ảnh trở thành một vector 1 chiều.

InceptionV3 và ResNet50: Đối với mô hình InceptionV3 và ResNet50, các bước tiền xử lý dữ liệu sau được thực hiện:

- Đọc các khung hình của mỗi video, trong đó hai khung hình liên tiếp được lấy cách nhau 15 khung hình.
- Sử dụng LabelBinarizer để gán nhãn cho các hình ảnh.
- Điều chỉnh kích thước của mỗi hình ảnh sao cho phù hợp với đầu vào của mô hình, với kích thước là (224, 224, 3).

Các bước tiền xử lý này giúp chuẩn bị dữ liệu phù hợp cho từng mô hình và tối ưu hóa hiệu suất của chúng trong quá trình nhận dạng hành động.

2.5 Data Augmentation

Để tăng cường dữ liệu, các kỹ thuật sau được sử dụng:

- **Rotation (Xoay)**: Áp dụng xoay với góc ngẫu nhiên từ -30 đến 30 độ để giúp mô hình học được sự biến đổi hình học của các hành động.
- **Zoom**: Áp dụng tăng hoặc giảm tỷ lệ thu phóng với giá trị ngẫu nhiên từ 0 đến 0.15 để đa dạng hóa các kích thước hình ảnh.
- **Width Shift (Dịch ngang)** và **Height Shift (Dịch dọc)**: Áp dụng dịch ngang và dọc với tỷ lệ từ -0.2 đến 0.2 để giúp mô hình học được sự di chuyển của các hành động.
- **Shear (Góc cắt)**[6]: Áp dụng cắt với góc ngẫu nhiên từ -15 đến 15 độ để giúp mô hình học được các biến đổi không gian của các hành động.
- **Horizontal Flip (Lật ngang)**: Áp dụng lật ngang với xác suất 50% để tạo sự đối xứng và đa dạng hóa dữ liệu.

Các kỹ thuật tăng cường dữ liệu này giúp tạo ra nhiều biến thể hình ảnh từ dữ liệu huấn luyện ban đầu, giúp mô hình học tập hiệu quả và khái quát hóa tốt hơn trong quá trình nhận dạng hành động[8].

3 Đánh giá thực nghiệm

3.1 Bộ dữ liệu

Bộ dữ liệu RLVS (Real Life Violence Situations) được sử dụng để thực hiện nhận dạng hành động với ba mô hình XGBoost, ResNet50 và InceptionV3.

Bộ dữ liệu RLVS bao gồm 1000 video chứa tình huống bạo lực và 1000 video không có bạo lực, được thu thập từ các video trên YouTube. Các video về tình huống bạo lực chứa nhiều tình huống thực tế trong các môi trường và điều kiện khác nhau. Các video không có bạo lực bao gồm các hành động như thể thao, ăn uống, đi bộ và nhiều hành động khác.

Bộ dữ liệu RLVS là một thách thức lớn trong việc nhận dạng hành động vì tính đa dạng và phức tạp của các tình huống trong cuộc sống thực. Sự đa dạng của bộ dữ liệu là cơ sở để xây dựng nên một mô hình nhận dạng tổng quát và ổn định trong nhiều tình huống thực tế khác nhau.

3.2 Quá trình Kiểm tra

Ở phần này, em sẽ giải thích về quá trình kiểm tra kết quả của ba mô hình XGBoost, ResNet50 và InceptionV3 trên Bộ dữ liệu RLVS. Trước tiên, bộ dữ liệu được chia thành hai tập: tập huấn luyện (trainset) và tập kiểm tra (testset). 80% dữ liệu được sử dụng cho tập huấn luyện và 20% dữ liệu được sử dụng cho tập kiểm tra.

Mô hình được huấn luận trên tập Train được sử dụng để dự đoán nhãn cho tập kiểm tra và tính toán các chỉ số đánh giá như độ chính xác (accuracy), precision, recall, và điểm F1 (F1-score) của các dự đoán trên tập kiểm tra.

Các chỉ số đánh giá hiệu suất được tính toán như sau:

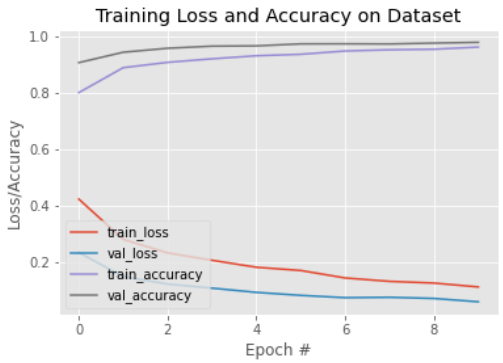
- **Độ chính xác (Accuracy)**: Tỉ lệ số lượng dự đoán chính xác trên tổng số dự đoán trên tập kiểm tra.
- **Precision**: Tỉ lệ số lượng dự đoán chính xác cho nhãn dương (bạo lực) trên tổng số dự đoán cho nhãn dương.
- **Recall**: Tỉ lệ số lượng dự đoán chính xác cho nhãn dương trên tổng số nhãn dương trong tập kiểm tra.

- **Điểm F1 (F1-score)**: Trung bình điều hòa giữa Precision và Recall, được tính bằng công thức: F1-score $= 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$.

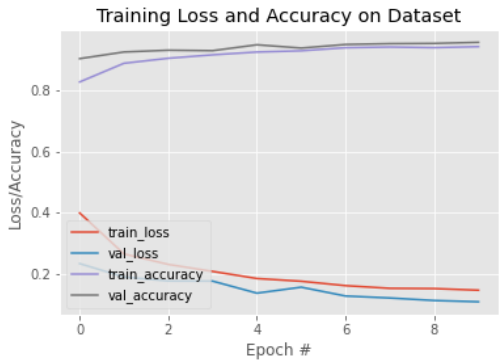
Kết quả của quá trình kiểm tra sẽ cung cấp thông tin quan trọng về hiệu suất của ba mô hình XGBoost, ResNet50 và InceptionV3 trong việc nhận dạng hành động trên Bộ dữ liệu RLVS.

3.3 Thiết lập tham số các mô hình

Trong mô hình XGBoost, các siêu tham số mặc định được sử dụng theo tài liệu hướng dẫn của XGBoost. Đối với mô hình ResNet50 và InceptionV3, các lớp Head được thêm vào theo cách đã đề cập ở phần trước. Trong đó tỉ lệ của lớp Dropout là 0.5. Mô hình Resnet50 sử dụng bộ tối ưu Adam với learning rate là 1×10^{-4}. Mô hình InceptionV3 sử dụng bộ tối ưu SGD với *momentum* là 0.9 và learning rate là 1×10^{-4}. Các lớp cơ sở được đóng băng và chỉ các lớp Head được huấn luyện. Sau 10 epochs, 2 mô hình hội tụ như hình dưới đây:



Hình 6: Độ chính xác và giá trị hàm mất mát của InceptionV3



Hình 7: Độ chính xác và giá trị hàm mất mát của ResNet50

3.4 Kết quả và thảo luận

Trong phần này, kết quả thu được từ ba mô hình: XGBoost, InceptionV3, và ResNet50 được đánh giá dựa trên các tiêu chí như độ chính xác (Accuracy), Precision, Recall, và điểm F1 (F1-score).

Mô hình	Precision	Recall	Điểm F1 (F1-score)	Độ chính xác(Accuracy)
XGBoost	0.95	0.94	0.94	0.94
InceptionV3	0.96	0.97	0.97	0.97
ResNet50	0.96	0.96	0.96	0.96

Bảng 1: Kết quả từ ba mô hình khác nhau

Như có thể thấy từ bảng trên, mô hình InceptionV3 cho kết quả tốt nhất với Precision, Recall, và điểm F1 lần lượt là 0.96, 0.97, và 0.97. Mô hình ResNet50 cũng cho kết quả tương tự với InceptionV3 nhưng có một sự khác biệt nhỏ về Recall và điểm F1. Trong khi đó, mô hình XGBoost có hiệu suất tương đối thấp hơn so với hai mô hình còn lại.

Dựa trên kết quả này, có thể rút ra rằng trong việc nhận diện hành động, mô hình InceptionV3 cho kết quả tốt nhất trong ba mô hình đã được thử nghiệm, cho thấy rằng đây là mô hình đáng tin cậy nhất trong việc nhận diện hành động trong các ứng dụng thực tế.

4 Kết luận và Công việc trong tương lai

Trong Project II này, em đã thực hiện nhận dạng hành động sử dụng ba mô hình XGBoost, ResNet50 và InceptionV3 trên Bộ dữ liệu RLVS (Real Life Violence Situations). Em đã sử dụng các phương pháp tiền xử lý dữ liệu và áp dụng kỹ thuật tăng cường dữ liệu để chuẩn bị dữ liệu huấn luyện cho từng mô hình. Sau đó, em thực hiện quá trình kiểm tra và tính toán các chỉ số đánh giá hiệu suất như độ chính xác, Precision, Recall và điểm F1 trên tập kiểm tra.

Kết quả thử nghiệm cho thấy mô hình InceptionV3 đạt được hiệu suất tốt nhất trong việc nhận dạng hành động trên Bộ dữ liệu RLVS. Mô hình này đã vượt trội so với mô hình XGBoost và Resnet50, cho thấy tính hiệu quả của việc sử dụng mô hình học sâu trong nhận dạng hành động.

Tuy nhiên, vẫn còn một số hạn chế trong Project lần này. Bộ dữ liệu RLVS, mặc dù đa dạng và phong phú, vẫn có thể chưa phản ánh hết các tình huống hành động trong cuộc sống thực. Điều này có thể ảnh hưởng đến tính tổng quát của mô hình trong một số tình huống đặc biệt. Đồng thời, việc tăng cường dữ liệu và tiền xử lý dữ liệu cũng có thể cần được điều chỉnh để cải thiện hiệu suất của mô hình.

Với những hạn chế và kết quả nghiên cứu này, em đề xuất một số hướng phát triển trong tương lai.

- Mở rộng bộ dữ liệu RLVS để bao gồm thêm các tình huống và hành động mới từ nhiều nguồn dữ liệu khác nhau, để cải thiện tính đại diện của bộ dữ liệu.
- Thử nghiệm các mô hình khác để giải quyết vấn đề đại diện mối quan hệ giữa các hình ảnh trong một video (CNN-LSTM, mô hình Transformer hoặc mô hình Graph Neural Network (GNN))

Tuy vậy, thử nghiệm các mô hình mới và mở rộng bộ dữ liệu cũng đặt ra nhiều thách thức về mặt tính toán và cần sự cân nhắc kỹ lưỡng trong việc điều chỉnh tham số. Điều này đòi hỏi em cần tiếp tục nghiên cứu và thử nghiệm để đảm bảo tính chính xác và hiệu quả của các mô hình trong thực tế.

References

[1] Almamon Rasool Abdali. "Data Efficient Video Transformer for Violence Detection". In: *2021 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT)*. 2021, pp. 195–199. DOI: 10.1109/COMNETSAT53002.2021.9530829.

[2] Tianqi Chen and Carlos Guestrin. "XGBoost". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, Aug. 2016. DOI: 10.1145/2939672.2939785. URL: https://doi.org/10.1145%2F2939672.2939785.

[3] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: (2015). arXiv: 1512.03385 [cs.CV].

[4] Hamid Mohammadi and Ehsan Nazerfard. "Video Violence Recognition and Localization Using a Semi-Supervised Hard Attention Model". In: (2022). arXiv: 2202.02212 [cs.CV].

[5] Nadia Mumtaz et al. "An Overview of Violence Detection Techniques: Current Challenges and Future Directions". In: (2022). arXiv: 2209.11680 [cs.CV].

[6] Luis Perez and Jason Wang. "The Effectiveness of Data Augmentation in Image Classification using Deep Learning". In: (2017). arXiv: 1712.04621 [cs.CV].

[7] Christian Szegedy et al. "Rethinking the Inception Architecture for Computer Vision". In: (2015). arXiv: 1512.00567 [cs.CV].

[8] Suorong Yang et al. "Image Data Augmentation for Deep Learning: A Survey". In: (2022). arXiv: 2204.08610 [cs.CV].