

**Nhận diện hình ảnh thời trang áp dụng
Multinomial Logistic Regression và Convolutional Neural Network**

Nguyễn Chí Dũng – 20002040

Đinh Tiến Dũng – 20002039

Nguyễn Hoàng Giang – 20002048

Group 05

01

ĐẶT VẤN ĐỀ

02

TRỰC QUAN HÓA
DỮ LIỆU

NỘI DUNG

03

XÂY DỰNG MÔ HÌNH

04

KẾT QUẢ

Phân loại hình ảnh đang là bài toán rất được quan tâm trong lĩnh vực trí tuệ nhân tạo. Để giải quyết bài toán này, nhiều mô hình khác nhau đã được phát triển, trong đó có hai mô hình thường được sử dụng là Convolutional Neural Network và Multinomial Logistic Regression.

Các mô hình này rất phù hợp cho bài toán phân loại hình ảnh vì:

- Mô hình Convolutional Neural Network có khả năng trích xuất đặc trưng của bức ảnh cực kỳ tốt và phù hợp cho bài toán phân loại hình ảnh với độ chính xác cao.
- Mô hình Multinomial Logistic Regression cũng là một mô hình phân loại phổ biến, sử dụng cho bài toán phân loại đa lớp.

Đề tài sử dụng bộ dữ liệu Fashion-MNIST để thực hiện quá trình huấn luyện và dự đoán. Đây là một trong những bộ dữ liệu tiêu chuẩn được áp dụng trong rất nhiều mô hình học máy với mục đích nhận diện hình ảnh. Với việc áp dụng đồng thời cả hai mô hình trên bộ dữ liệu Fashion-MNIST, chúng ta sẽ có một cái nhìn toàn diện và so sánh được độ hiệu quả giữa các mô hình.

Fashion-MNIST (Fashion – Modified National Institute of Standards and Technology) là bộ dữ liệu gồm các hình ảnh thời trang thuộc về công ty thương mại điện tử Zalando.

Zalando tạo ra bộ dữ liệu Fashion-MNIST nhằm đóng vai trò thay thế trực tiếp cho bộ dữ liệu MNIST ban đầu về chữ viết tay để đo điểm chuẩn cho các thuật toán học máy. Nó chia sẻ cùng kích thước hình ảnh, cấu trúc phân tách huấn luyện và kiểm tra.

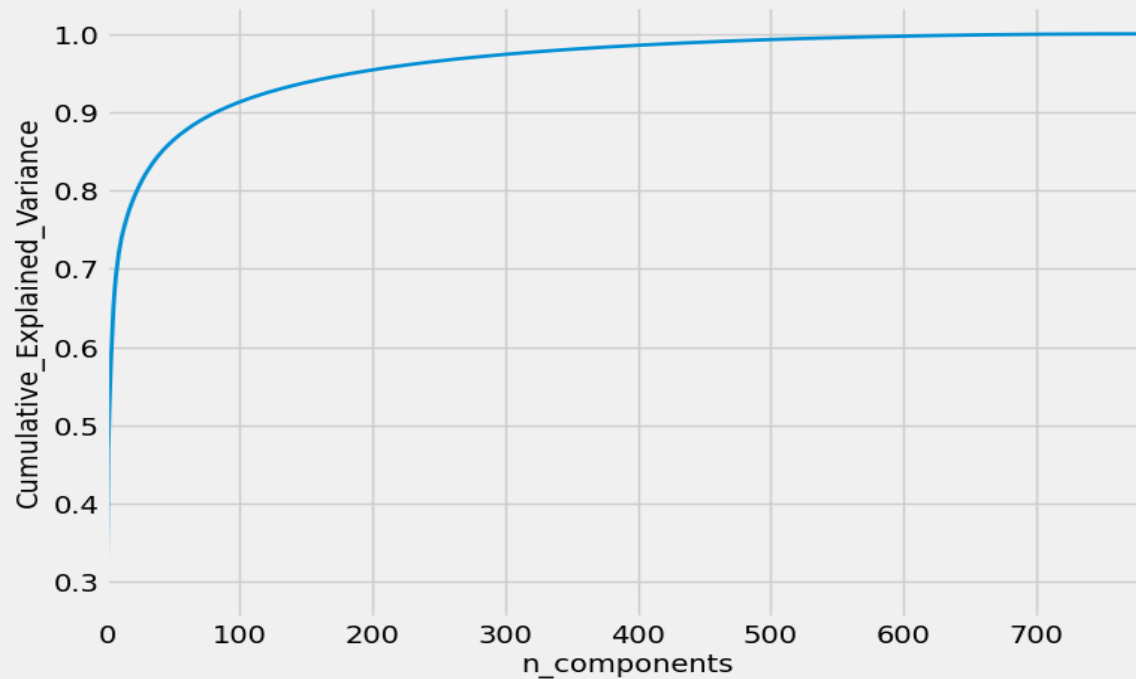
Các hình ảnh có kích thước 28x28 pixels, với mỗi pixel mang một giá trị cho biết độ sáng hoặc độ tối, chỉ số càng cao thì càng tối. Giá trị pixel là một số nguyên từ 0 đến 255.

● BỘ DỮ LIỆU FASHION-MNIST



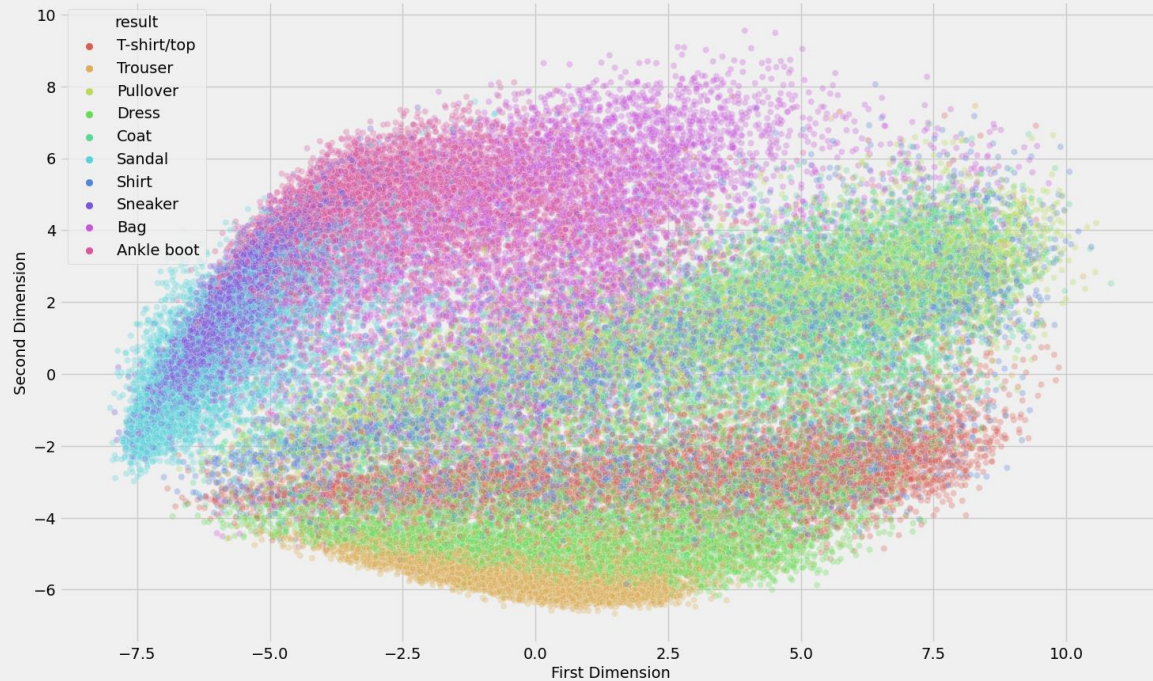
30 hình ảnh của bộ dữ liệu huấn luyện

● TRỰC QUAN HÓA DỮ LIỆU



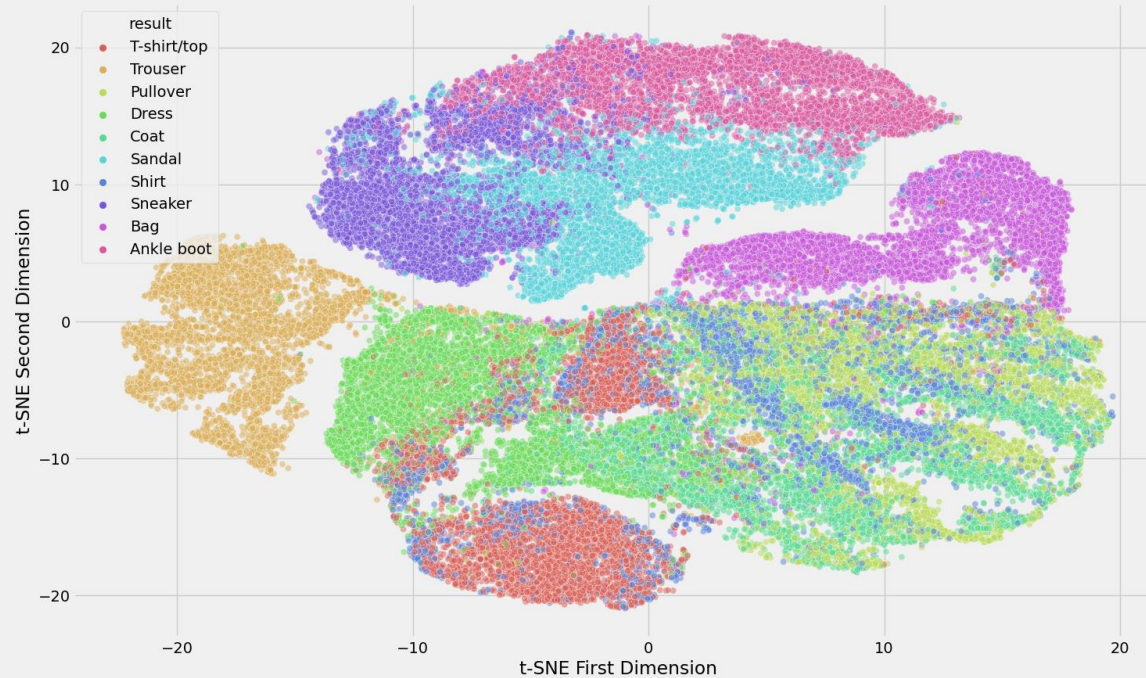
Đồ thị thể hiện sự hiệu quả khi giảm chiều với số chiều từ 0 đến 784

● TRỰC QUAN HÓA DỮ LIỆU



Đồ thị phân tán các nhãn theo PCA

● TRỰC QUAN HÓA DỮ LIỆU



Đồ thị phân tán các nhãn theo t-SNE

● TRỰC QUAN HÓA DỮ LIỆU

K-Means là thuật toán phân cụm thuộc loại học không giám sát. Ý tưởng của thuật toán phân cụm K-Means là phân chia một bộ dữ liệu thành các cụm khác nhau, trong đó số lượng cụm được cho trước là k . Công việc phân cụm được xác lập dựa trên nguyên lý: Các điểm dữ liệu trong cùng một cụm phải có cùng một số tính chất nhất định.

Các bước thực hiện K-Means:

Bước 1: Xác định k cụm ($k > 0$)

Bước 2: Khởi tạo ngẫu nhiên k tâm (Mỗi tâm tương ứng với một cụm)

Bước 3: Gán mỗi điểm dữ liệu vào cụm gần nhất (Dựa vào khoảng cách từ điểm đó tới tâm cụm gần nhất)

Bước 4: Cập nhật tâm cụm mới của mỗi cụm

Bước 5: Lặp lại bước 3 và bước 4 cho đến khi tâm cụm không thay đổi.

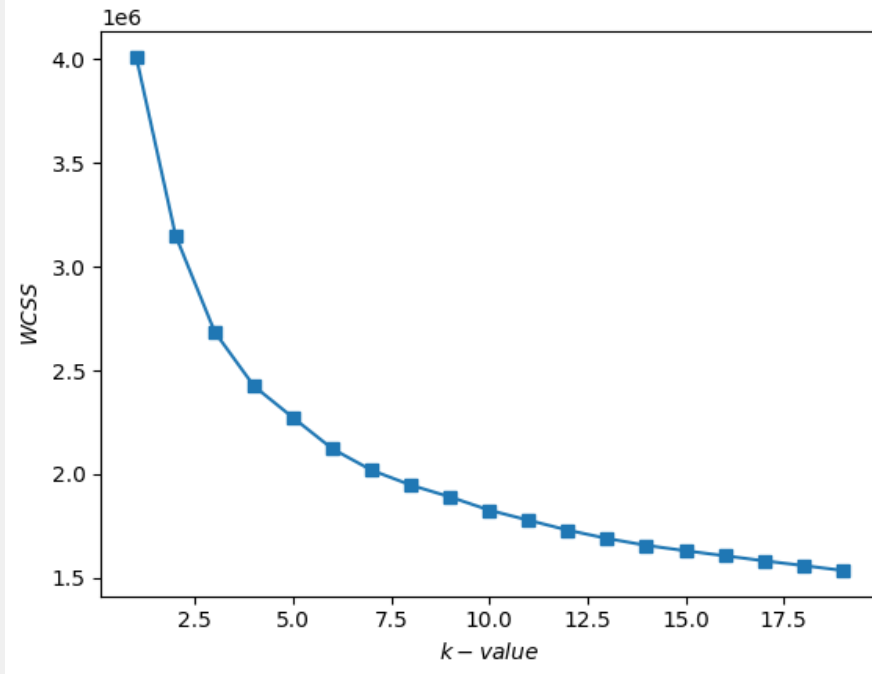
● PHÂN CỤM K-Means

Chọn số cụm trong K-Means

*Phương pháp Elbow

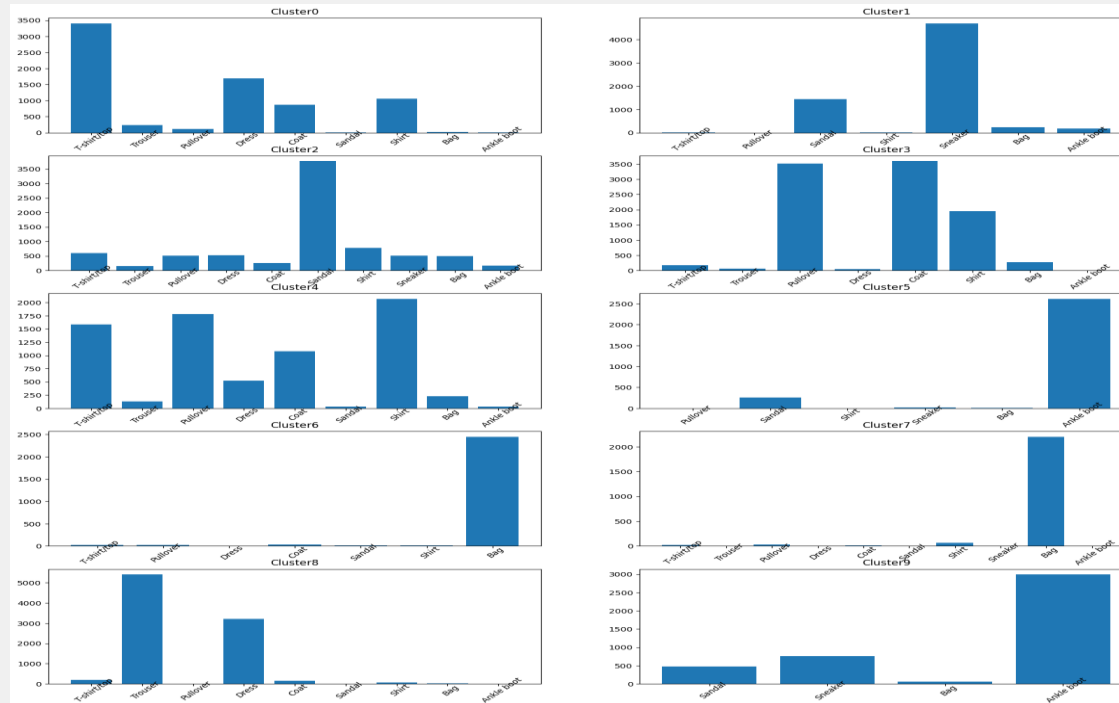
- Sử dụng thuật toán K-Means với các giá trị số lượng cụm khác nhau.
- Với mỗi số lượng cụm, tính tổng bình phương khoảng cách của các điểm đến tâm cụm tương ứng.
- Vẽ đường cong tổng bình phương khoảng cách (Within-Cluster Sum of Square - WCSS) với số lượng cụm trên trục x.
- Xác định số lượng cụm tối ưu bằng cách chọn điểm “khủy tay” của đường cong, tức là số lượng cụm mà sau đó việc tăng số lượng cụm không đem lại sự giảm thiểu đáng kể về tổng bình phương khoảng cách.

● PHÂN CỤM K-Means



Đồ thị chọn số cụm cho bộ dữ liệu Fashion-MNIST

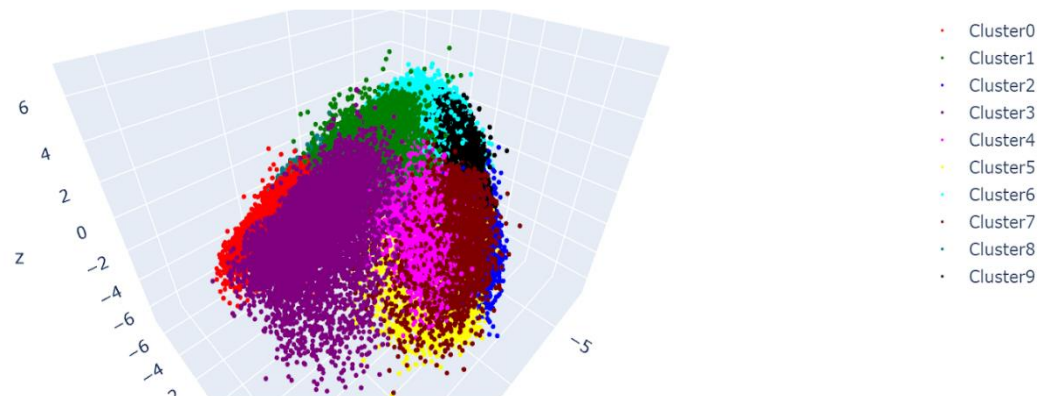
● PHÂN CỤM K-Means



Đồ thị số lượng theo từng nhãn trong mỗi cụm

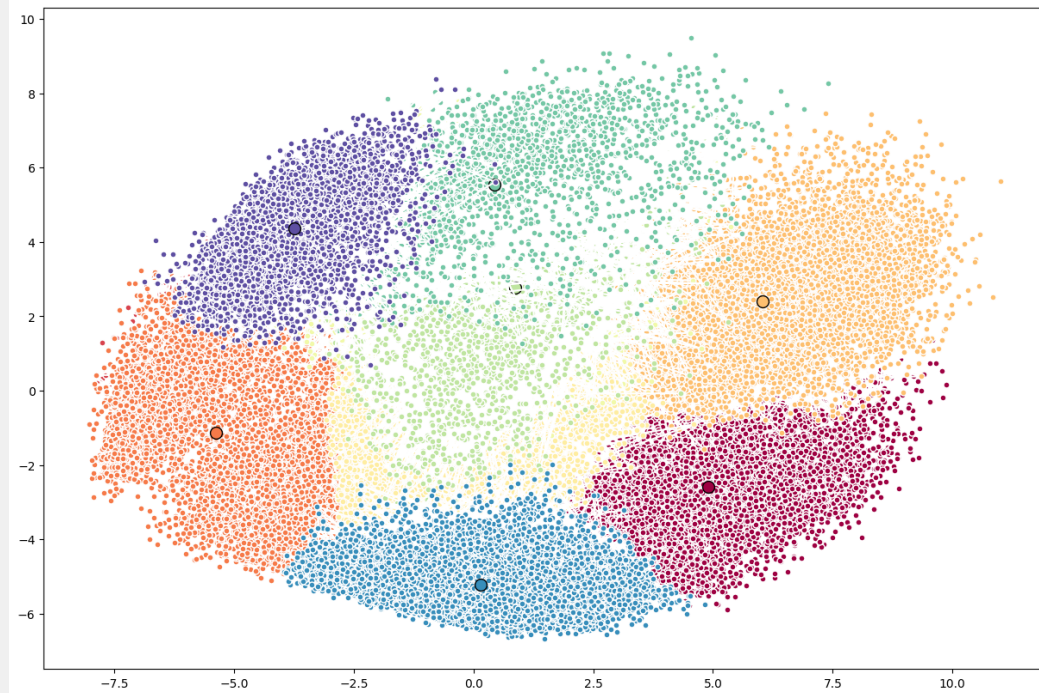
● PHÂN CỤM K-Means

Cluster Visualisation



Đồ thị phân cụm 3D

● PHÂN CỤM K-Means



Đồ thị phân cụm 2D

● PHÂN CỤM K-Means

Sử dụng thư viện: numpy, pandas, tensorflow, matplotlib, seaborn, sklearn, keras.

Bộ dữ liệu training: 80% bộ dữ liệu training gốc

Bộ dữ liệu validation: 20% bộ dữ liệu training gốc

Bộ dữ liệu testing: Toàn bộ dữ liệu thuộc bộ dữ liệu testing gốc

Đầu vào mô hình: Dữ liệu dưới dạng số, theo chuẩn min-max

Đầu ra mô hình: Các nhãn được đánh số từ 0 đến 9

Chỉ số đánh giá: Accuracy.

● XÂY DỰNG MÔ HÌNH

Multinomial Logistic Regression

Hồi quy logistic đa thức hay còn gọi là hồi quy softmax là trường hợp tổng quát của hồi quy logistic để xử lý trường hợp có nhiều hơn hai nhãn. Trong hồi quy logistic, nhãn được giả thuyết ở dạng nhị phân $y^{(i)} \in \{0,1\}$.

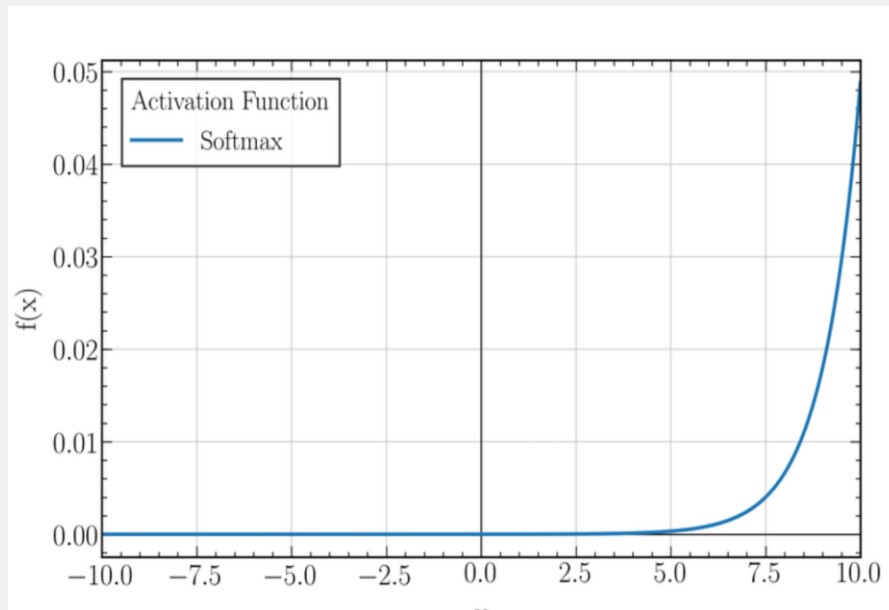
Trong khi đó, hồi quy softmax cho phép giải quyết trường hợp nhãn $y^{(i)} \in \{1,\dots,k\}$ với k là số nhãn.

Softmax

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

Trong đó:

- z là vector có k chiều
- i là lớp được xét
- k là tổng số lớp.



● XÂY DỰNG MÔ HÌNH

Hàm tổn thất Cross Entropy

Hàm Cross-entropy (Negative Log Likelihood) được sử dụng để đo lường mức độ sai lệch giữa giá trị dự đoán của mô hình và giá trị thực tế. Đầu vào của hàm Cross-entropy là hai phân phối xác suất: Phân phối dự đoán của mô hình và phân phối thực tế của dữ liệu đích. Công thức của hàm Cross-entropy như sau:

$$J(W; x_i, y_i) = - \sum_{j=1}^c y_{ji} \log(a_{ji})$$

Với y_{ij} và a_{ij} lần lượt là phần tử thứ j của vector y_i và a_i .

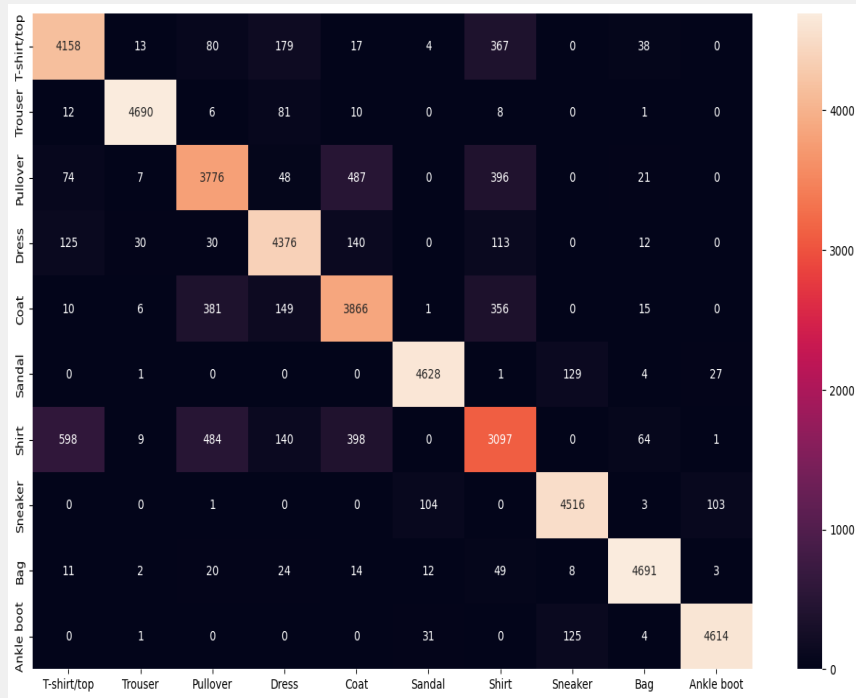
Giải bài toán tối ưu $J(\theta) \rightarrow \min_{\theta}$

Schochastic Gradient Descent - SGD

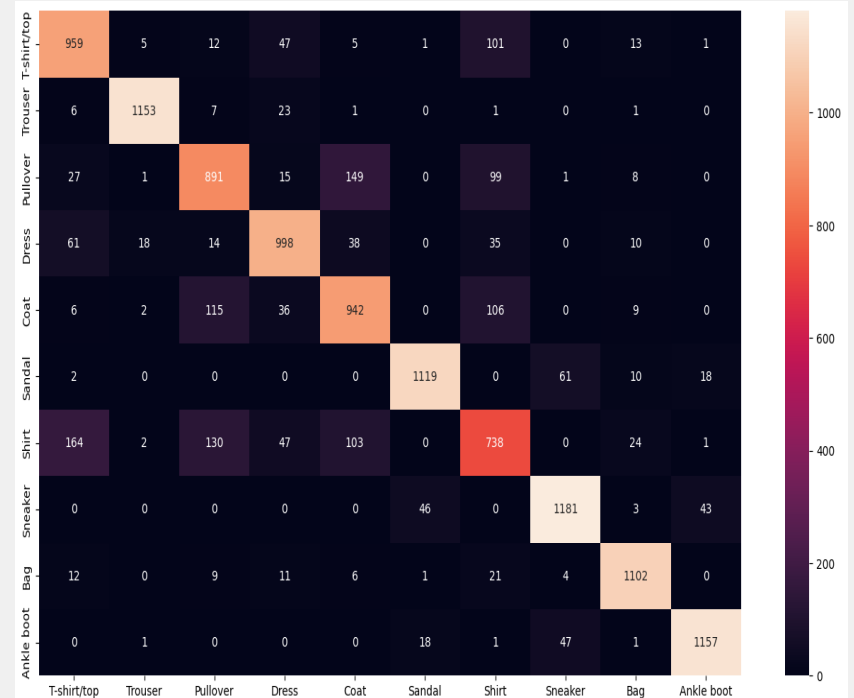
1. Đầu vào: Tham số học $\eta > 0$; (x_n, y_n) . Khởi tạo: Chọn $\theta \in \mathbb{R}^{d \times c}$ bất kỳ ở bước $st = 0$;
2. Tráo đổi ngẫu nhiên thứ tự các cặp dữ liệu (x_n, y_n) . Lần lượt với $n = 1, 2, \dots, N$, thực hiện:
3. Tính $\frac{\partial J_n(\theta)}{\partial \theta}$
4. Cập nhật tham số:

$$\theta = \theta - \eta \cdot \frac{\partial J_n(\theta)}{\partial \theta}$$

5. Nếu đạt điều kiện dừng, kết thúc. Ngược lại quay lại bước 2.

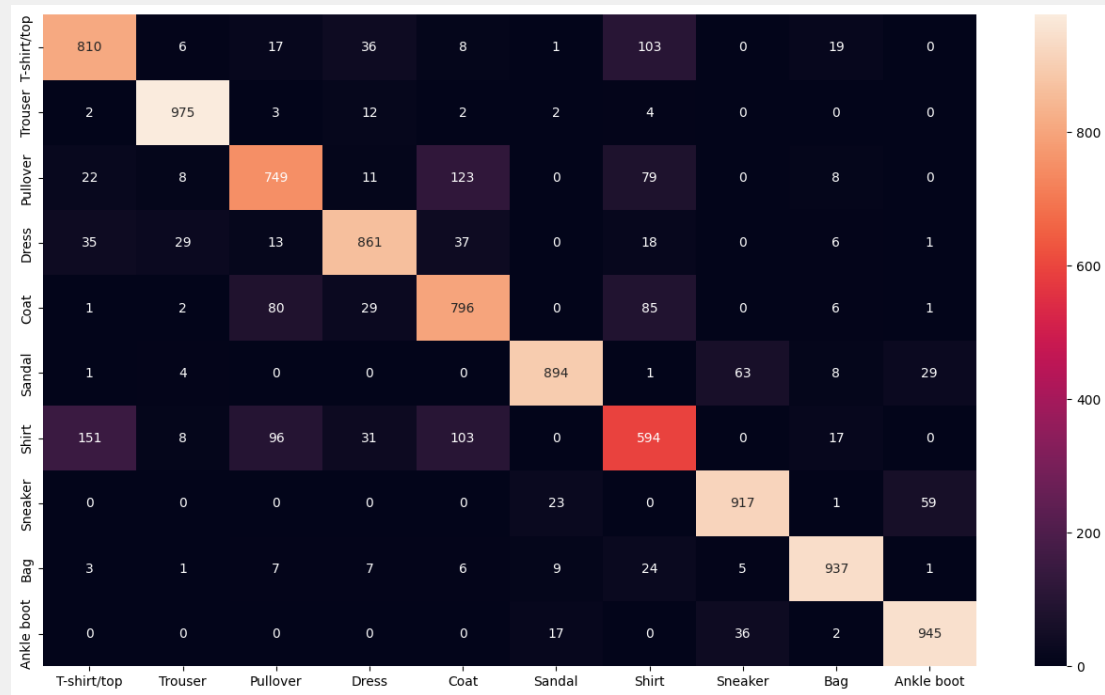


Training: 88.36%



Validation: 85.33%

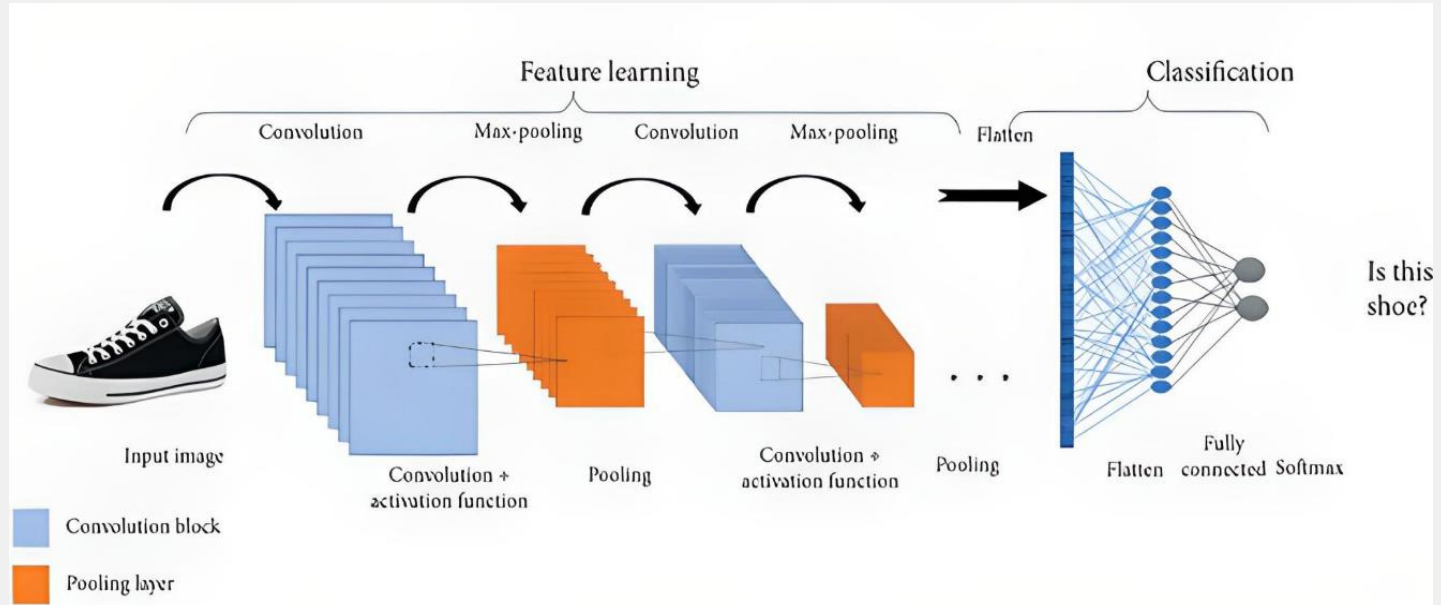
● KẾT QUẢ



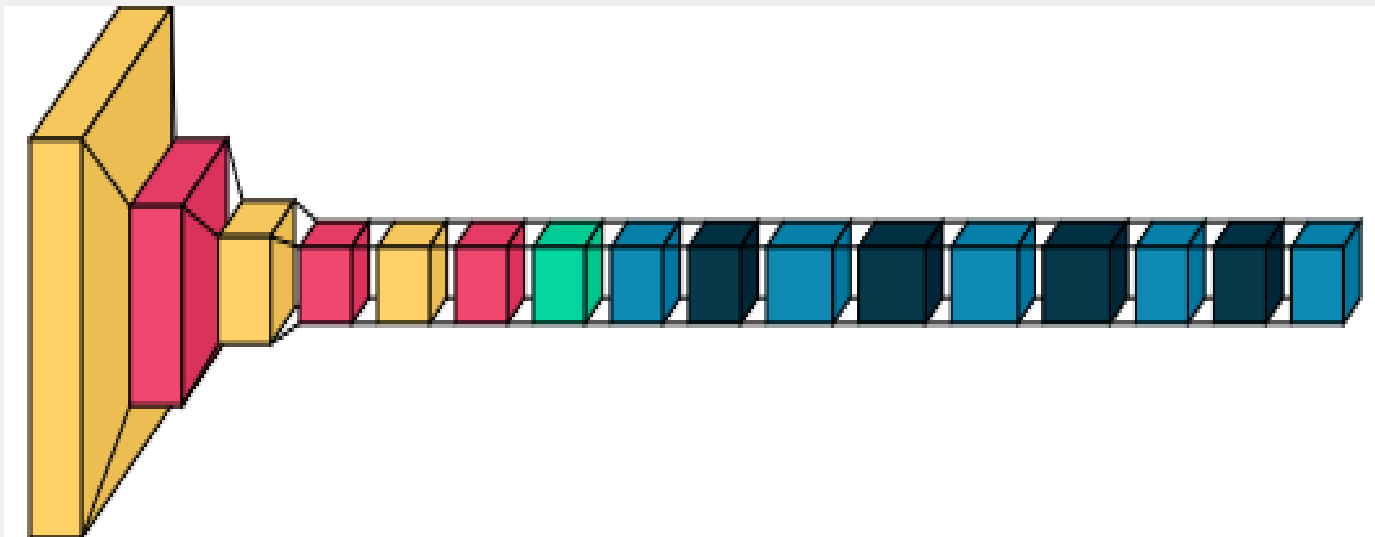
Testing: 84.59%

● KẾT QUẢ

Convolutional Neural Network

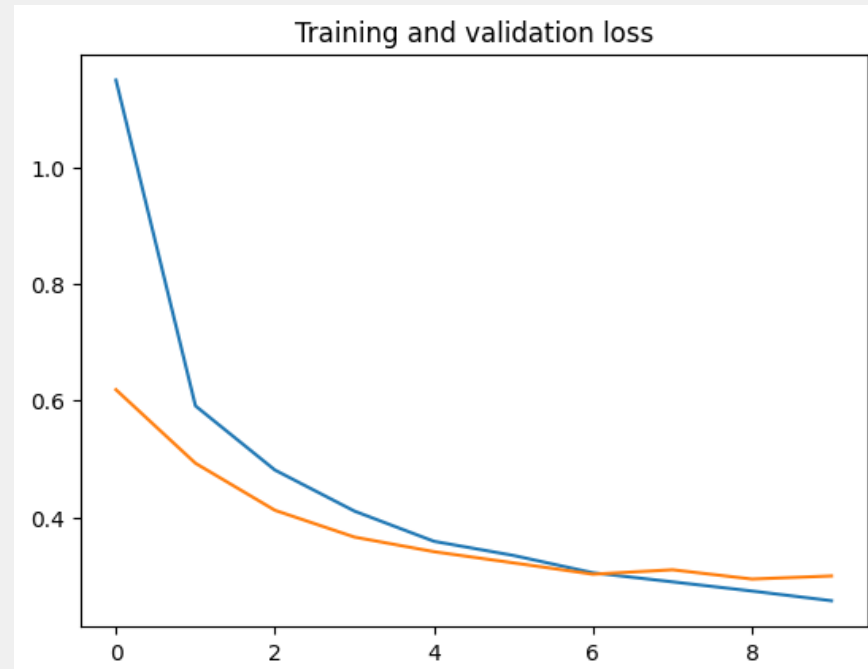
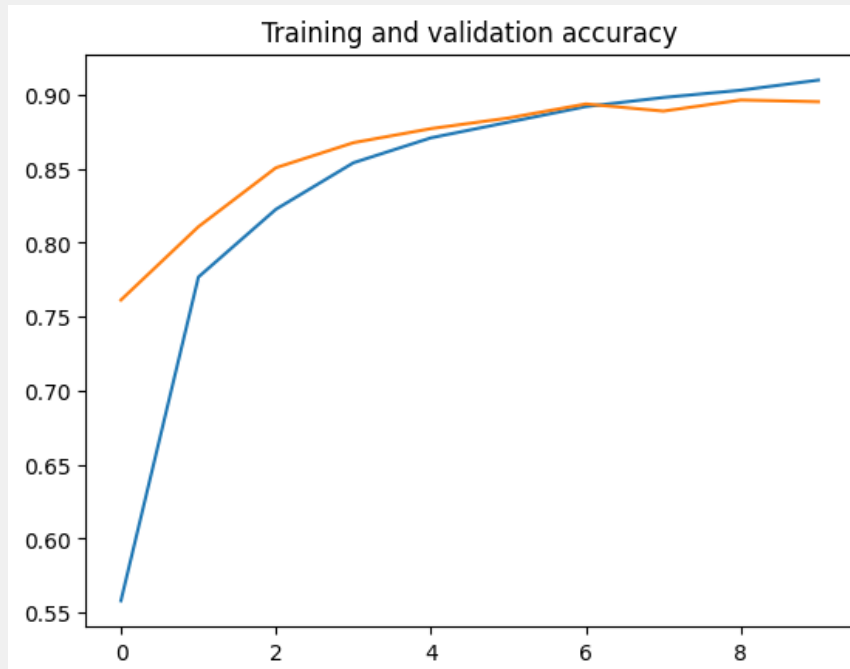


● XÂY DỰNG MÔ HÌNH



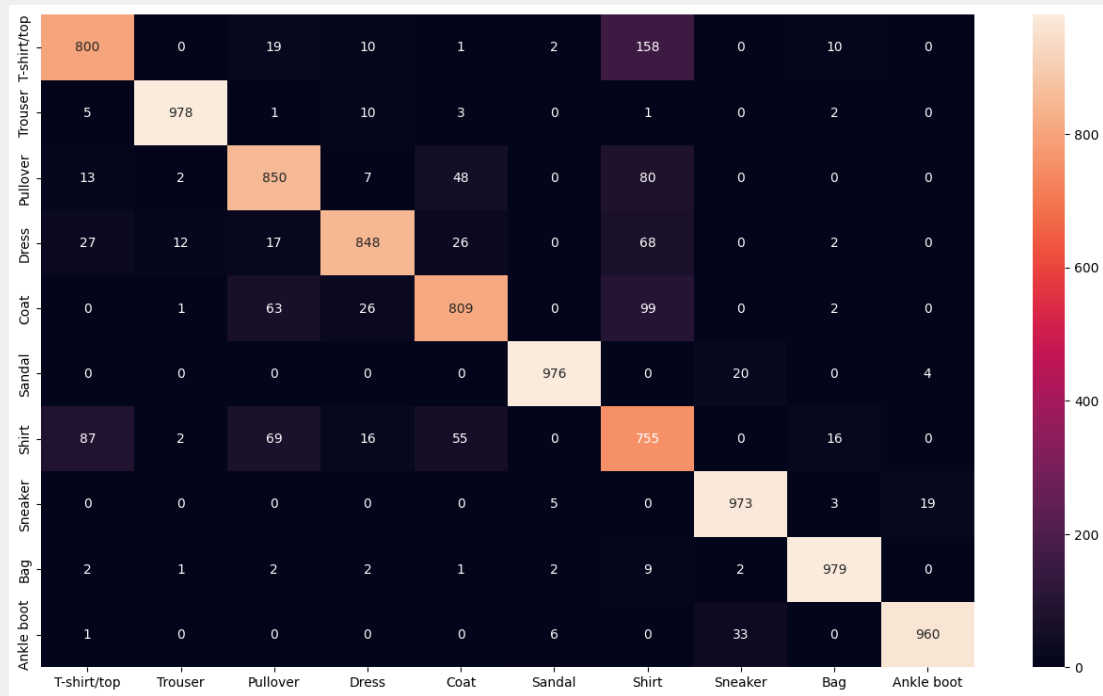
3 lớp Convolution – 3 lớp Maxpooling – 1 lớp Flatten – 5 lớp Dense – 4 lớp Dropout

● XÂY DỰNG MÔ HÌNH



Kết quả chạy của mô hình trên tập training và validation với 10 epochs

● KẾT QUẢ



Test: 89.27%

● KẾT QUẢ

- Done by:
Nguyễn Chí Dũng - 20002040
Đinh Tiến Dũng - 20002039
Nguyễn Hoàng Giang - 20002048
- *Do you have any questions?*
nguyenchidung_t65@hus.edu.vn
+84 947 985 594

Group 05



Thanks !