**BÁO CÁO TUẦN 5 ĐỒ ÁN MÔN HỌC**

Tên đồ án: “Ứng dụng các mô hình máy học vào bài toán phân loại hoạt động của người dùng, trên các thiết bị đeo tay theo dõi sức khỏe.”

Thành viên thực hiện:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | HỌ VÀ TÊN | MSSV | ĐIỆN THOẠI | EMAIL |
| 1 | Nguyễn Ngọc Minh | 19520165 | 0585115056 | 19520165@gm.uit.edu.vn |

1. Nội dung công việc trong tuần

Train model, đánh giá độ chính xác và tối ưu hóa model cho MCU có dung lượng nhớ ít.

1. Báo cáo thực hiện
2. Chuẩn bị dữ liệu

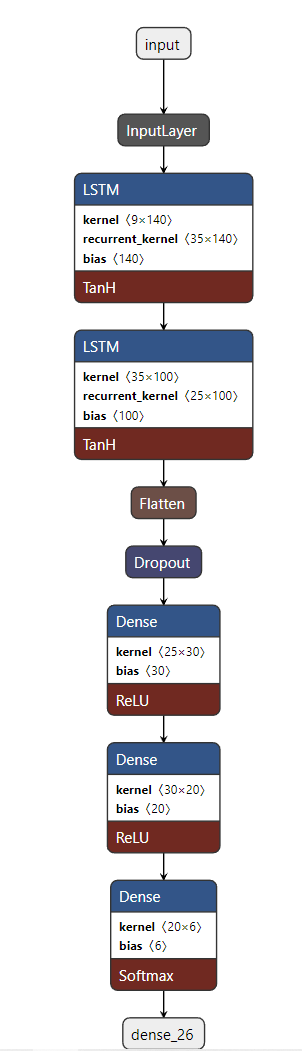
|  |
| --- |
| #Các feature  INPUT\_SIGNAL\_TYPES = [      "body\_acc\_x\_",      "body\_acc\_y\_",      "body\_acc\_z\_",      "body\_gyro\_x\_",      "body\_gyro\_y\_",      "body\_gyro\_z\_",      "total\_acc\_x\_",      "total\_acc\_y\_",      "total\_acc\_z\_"  ]  # Các lớp phân loại  LABELS = [      "WALKING",      "WALKING\_UPSTAIRS",      "WALKING\_DOWNSTAIRS",      "SITTING",      "STANDING",      "LAYING"  ] |

|  |
| --- |
| trainX, trainy, testX, testy = load\_dataset(folder)  >>> trainX.shape: (7352, 128, 9)  >>> trainy.shape: (7352, 6)  >>> testX.shape: (2947, 128, 9)  >>> testy.shape: (2947, 6) |

1. Tiến hành training, đánh giá và tối ưu hóa:

* Mục tiêu: độ chính xác > 90% trên tập test, và số lượng param cần học nhỏ nhất có thể.

1. Phiên bản 1:



Layer (type) Output Shape Param #

=================================================================

lstm\_27 (LSTM) (None, 128, 35) 6300

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

lstm\_28 (LSTM) (None, 25) 6100

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

flatten\_8 (Flatten) (None, 25) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_8 (Dropout) (None, 25) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_24 (Dense) (None, 30) 780

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_25 (Dense) (None, 20) 620

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_26 (Dense) (None, 6) 126

=================================================================

Total params: 13,926

Trainable params: 13,926

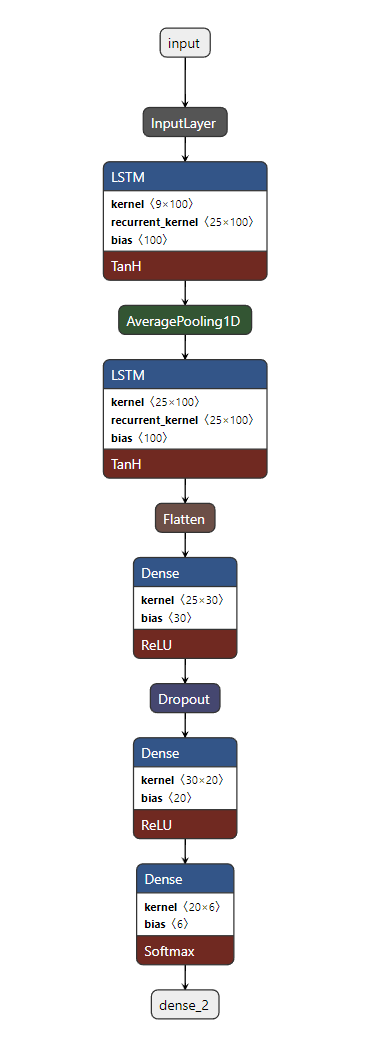
Non-trainable params: 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

93/93 [==============================] - 3s 24ms/step - loss: 0.3793 - accuracy: 0.9138

[0.37927448749542236, 0.913810670375824]

* Độ chính xác: 91,3%
* Số param: 13926
* Loss: 0.38

1. Phiên bản 2

Layer (type) Output Shape Param #

=================================================================

lstm (LSTM) (None, 128, 25) 3500

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

average\_pooling1d (AveragePo (None, 25, 25) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

lstm\_1 (LSTM) (None, 25) 5100

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

flatten (Flatten) (None, 25) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense (Dense) (None, 30) 780

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout (Dropout) (None, 30) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_1 (Dense) (None, 20) 620

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_2 (Dense) (None, 6) 126

=================================================================

Total params: 10,126

Trainable params: 10,126

Non-trainable params: 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

93/93 [==============================] - 2s 16ms/step - loss: 0.4170 - accuracy: 0.9223

[0.4169643223285675, 0.9222938418388367]

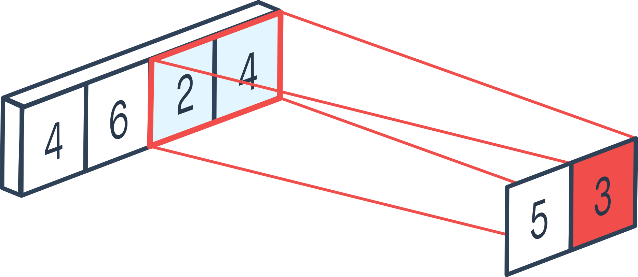
* Độ chính xác: 92, 23%
* Số param: 10126
* Loss: 0.41

1. Kết luận

* Mô hình 2 cho kết qua tốt hơn và tối ưu bộ nhớ sử dụng hơn nên ta chọn mô hình 2
* Giải thích các layer:

+ LSTM: thích hợp với cái input dạng time-series, mặc dù khiến thời gian train lâu hơn.

+ AveragePooling1D: Lấy giá trị trung bình trên từng feature với window size

 là 5. Loại bỏ các nhiễu tốt hơn.

+ Flatten: kéo phẳng dữ liệu trước khi cho vào lớp kế tiếp

+ ReLU: Trong mạng nơ ron không thể thiếu những lớp giúp nó học những thứ tuyến tính và phi tuyến tính.

Ảnh có chứa văn bản, đồng hồ

Mô tả được tạo tự động+ Softmax: Tất nhiên phải có trong bài toán phân loại nhiều lớp. Được đặt ở layer cuối cùng của các model. Với đầu ra là chỉ số xác suất với các class.

1. Ước lượng thời gian chạy realtime, ước lượng độ trễ, chiếm dụng bộ nhớ

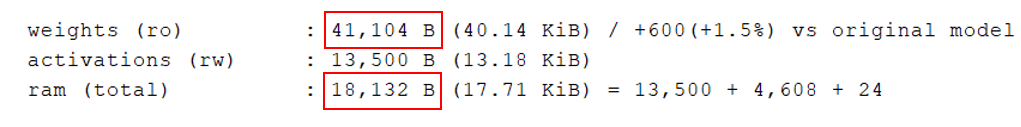
* Sử dụng công cụ STM32CubeAI
* Thông số CPU: Tần số sử dụng 100MHz, Ram 128KiB, Rom 512KiB.

Ảnh có chứa bàn

Mô tả được tạo tự động



* Thời gian xử lý 100ms



* Rom sử dụng 40.1 KiB (8%)
* Ram sử dụng 17.8 KiB (13%)
* Mô hình đạt yêu cầu về tốc độ cũng như bộ nhớ sử dụng

1. Công việc tuần kế tiếp:

* Tuần 6+7: Đưa mô hình xuống MCU với input được lấy mẫu thời gian thực từ sensor, tối ưu và đánh giá tốc độ đáp ứng.