TRƯỜNG ĐẠI HỌC CẦN THƠ
TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

NIÊN LUẬN CƠ SỞ NGÀNH

NGÀNH HỆ THỐNG THÔNG TIN

Đề tài

NGHIÊN CỨU MÔ HÌNH MÁY HỌC ĐỂ NHẬN DẠNG GIỌNG THẬT VÀ GIỌNG GIẢ

Sinh viên: Nguyễn Phi Nhân

Mã số: B2203461

Khóa: K48

Cần Thơ, 04/2025

TRƯỜNG ĐẠI HỌC CẦN THƠ
TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

NIÊN LUẬN CƠ SỞ NGÀNH

NGÀNH HỆ THỐNG THÔNG TIN

Đề tài

NGHIÊN CỨU MÔ HÌNH MÁY HỌC ĐỂ NHẬN DẠNG GIỌNG THẬT VÀ GIỌNG GIẢ

Sinh viên: Nguyễn Phi Nhân

Mã số: B2203461

Khóa: K48

Cần Thơ, 04/2025

LỜI CẢM ƠN

Trong quá trình học tập và nghiên cứu tại trường Đại học Cần Thơ, đây là lần đầu tiên em được thử sức mình trong việc vận dụng kiến thức học tập để ứng dụng vào một đề tài mang hướng thực tiễn và có thể nghiên cứu này là tiền đề để phát triển thành một dự án lớn hỗ trợ cộng đồng lớn hơn trong tương lai.

Để có thể nghiên cứu và phát triển đề tài em xin gửi lời cảm ơn chân thành nhất đến thầy Nguyễn Thanh Hải, cảm ơn thầy đã tận tình hỗ trợ, định hướng và giúp đỡ em rất nhiều trong suốt quá trình nghiên cứu đề tài, cảm ơn thầy đã dành nhiều thời gian, tâm huyết và công sức hỗ trợ em trong khoảng thời gian qua.

Bên cạnh những kết quả đã đạt được, đề tài vẫn có nhiều thiếu sót. Rất mong quý thầy cô thông cảm, mong quý thầy cô chỉ bảo, góp ý cho em, vì mỗi ý kiến đóng góp của quý thầy cô đều rất đáng trân trọng và là những kinh nghiệm, kiến thức có thể giúp em hoàn thiện bản thân mình hơn.

Bằng tất cả sự chân thành, một lần nữa cảm ơn mọi người, xin gửi lời chúc sức khỏe và mong cho mọi điều tốt đẹp sẽ đến với mọi người trong tương lai.

Trân trọng!

Cần Thơ, ngày 07 tháng 04 năm 2025

Tác giả

Nguyễn Phi Nhân

MỤC LỤC

DANH MỤC HÌNH

DANH MỤC BẢNG

DANH MỤC TỪ CHUYÊN NGÀNH

Viết tắt	Giải thích
DMF	Kỹ thuật phân rã ma trận sâu (Deep Matrix Factorization)
NLP	Xử lý ngôn ngữ tự nhiên (Natural Language Processing)
CNN	Mạng nơ-ron tích chập (Convolutional Neural Network)
LSTM	Mạng nơ-ron bộ nhớ ngắn dài (Long Short-Term Memory)
RNN	Mạng nơ-ron hồi tiếp (Recurrent Neural Network)
ReLU	Hàm kích hoạt tuyến tính (Rectified Linear Unit)
FC	Lớp kết nối đầy đủ (Fully Connected)

CHƯƠNG 1. GIỚI THIỆU VÀ MÔ TẢ BÀI TOÁN

1.1. Mục tiêu đề tài

Giọng giả (Deep fake voice) hay giọng cho trí tuệ nhân tạo tạo ra là một phần của trí thông minh nhân tạo, một hiện tưởng nổi lên sau sự thành công của Chatgpt (một chatbot do công ty OpenAI của Mỹ phát triển), kéo theo đó là một kỷ nguyên mới nơi trí thông minh nhân tạo được sử dụng với nhiều mục đích. Giọng giả có thể tạo ra âm thanh bắt chước giọng nói của con người một cách thuyết phục. Bằng cách tận dụng các kỹ thuật học máy hiện đại, đặc biệt là học sâu (Deep learning) các hệ thống này phân tích và tái tạo các sắc thái của lời nói con người một cách chính xác từ âm điệu, cao độ, tông giọng và phong cách nói chuyện. Giọng nói giả sau khi được tổng hợp khiến người nghe không thể phân biệt được đâu là giọng thật của người nói và đâu là giọng nói được tái tạo bởi trí thông minh nhân tạo.

Trong bối cảnh công nghệ thông tin phát triển như vũ bão, ranh giới giữa thực và ảo ngày càng trở nên mong manh, đặc biệt là trong lĩnh vực âm thanh. Sự xuất hiện của công nghệ tạo giọng nói giả mạo (Deep fake voice) đã đặt ra những thách thức lớn, đòi hỏi chúng ta phải nâng cao cảnh giác và có biện pháp phòng ngừa hiệu quả. Việc phân biệt giọng nói thật và giả không chỉ giúp bảo vệ cá nhân khỏi các hành vi lừa đảo tinh vi, mà còn góp phần duy trì sự tin cậy trong giao tiếp trực tuyến.

Các đối tượng xấu có thể lợi dụng công nghệ này để giả mạo giọng nói của người thân, bạn bè, đồng nghiệp hoặc các nhân vật có uy tín nhằm mục đích lừa đảo, chiếm đoạt tài sản, hoặc lan truyền thông tin sai lệch. Điều này không chỉ gây thiệt hại về tài chính mà còn ảnh hưởng nghiêm trọng đến danh dự và uy tín của nạn nhân.

Hơn nữa, trong một thế giới mà các thiết bị và hệ thống điều khiển bằng giọng nói ngày càng trở nên phổ biến, việc đảm bảo tính xác thực của các lệnh thoại là vô cùng quan trọng. Nếu không có biện pháp bảo vệ phù hợp, các hệ thống này có thể bị lợi dụng để thực hiện các hành vi trái phép, gây ra những hậu quả khó lường.

Chính vì vậy, việc phát triển các công cụ và kỹ thuật phát hiện giọng nói giả mạo là vô cùng cần thiết. Điều này không chỉ giúp bảo vệ quyền riêng tư và danh tính của cá nhân, mà còn góp phần xây dựng một môi trường giao tiếp trực tuyến an toàn và đáng tin cậy. Đồng thời, việc nâng cao nhận thức của cộng đồng về những rủi ro tiềm ẩn của công nghệ này cũng đóng vai trò quan trọng trong việc phòng ngừa và giảm thiểu tác hại của các hành vi lừa đảo bằng giọng nói giả mạo.

1.2. Mô tả chi tiết đề tài

Đề tài "***Nhận diện giọng thật và giọng giả***" mục tiêu của đề tài sẽ xây dựng một mô hình máy học có thể phân loại các dữ liệu âm thanh đầu vào và cho biết dữ liệu âm thanh đó bản chất là giọng của con người hay giọng được tái tạo bởi các mô hình máy học. Với đề tài này, trước tiên ta cần xử lý tính hiệu âm thanh đầu vào, sau đó tiến hành

nghiên cứu xây dựng huấn luyện mô hình Convolution Neural Network (CNN) để phân loại âm thanh dựa trên tính hiệu đầu vào đã được xử lý.

Các mục tiêu cụ thể như sau:

- Nghiên cứu phương pháp xử lý tính hiệu âm thanh đầu vào, đảm bảo dữ liệu đầu vào và dữ liệu sau khi được xử lý của hai loại âm thanh có thể đáp ứng được cho quá trình huấn luyện và đánh giá mô hình.

- Nghiên cứu và xây dựng mô hình phân loại giọng thật và giọng giả dựa trên tập dữ liệu âm thanh bằng mô hình Convolution Neural Network (CNN) có độ chính xác cao và thời gian ngắn. Tiến hành kiểm tra và đnáh giá mô hình ở nhiều độ đó khác nhau.

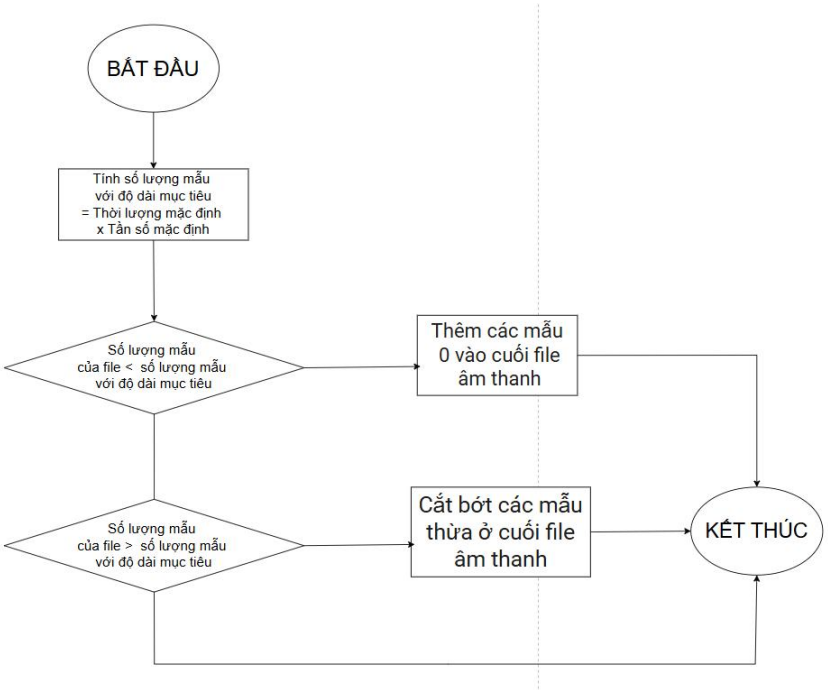
1.3. Hướng tiếp cận giải quyết của đề tài

1.3.1. Phương pháp tiền xử lý dữ liệu

Với dữ liệu đầu vào là các tệp âm thanh ta sẽ tiến hành các bước như sau

Bước 1: Gom nhóm các tệp âm thanh về cùng một loại.

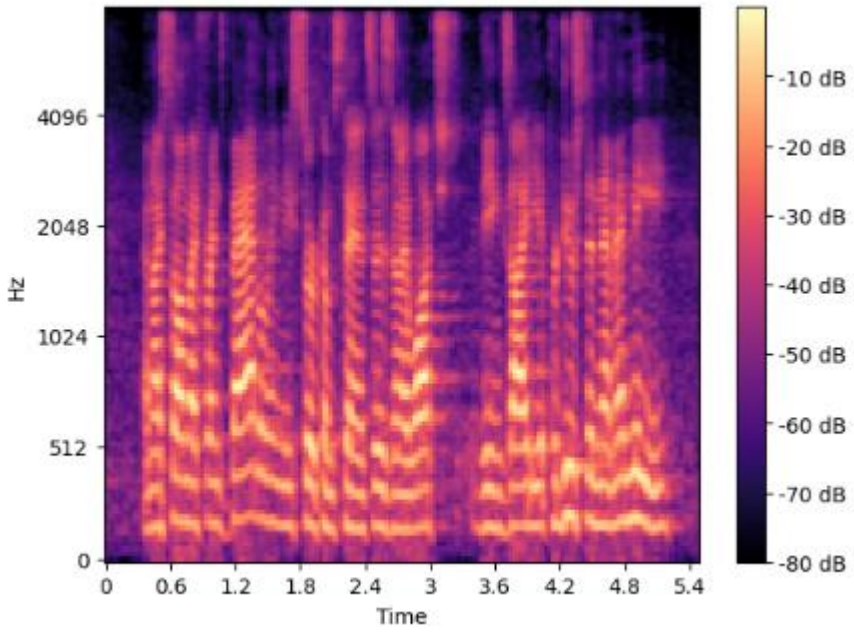
Bước 2: Tại mỗi nhóm, tiến hành điều chỉnh thời lượng âm thanh từng tệp về cùng một kích thước.



Hình 1.1. Lưu đồ tiền sử lý dữ liệu.

Bước 3: Gán nhãn cho từng tệp trong mỗi nhóm, đối với các tệp âm thanh giọng thật sẽ gán nhãn 1, các tệp âm thanh giọng giả sẽ gán nhãn 0.

Bước 4: Chuyển dữ liệu âm thanh từ 1D, sang phổ Mel Spectrogram



Hình 1.2. Dữ liệu âm thanh dưới dạng 2D

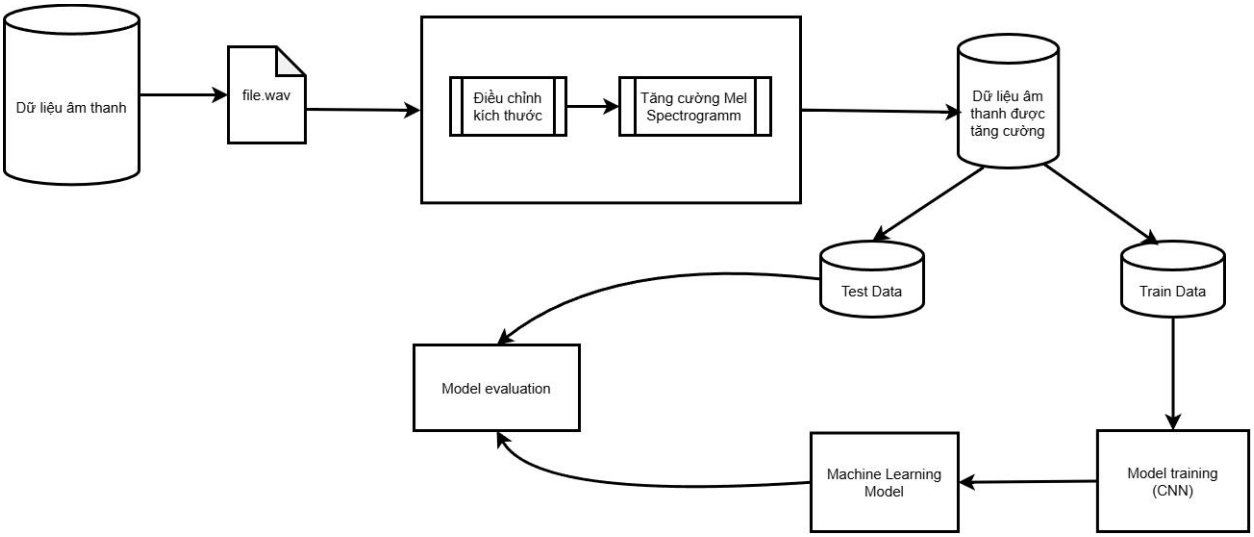
1.3.2. Các mô hình có thể sử dụng cho để tài

- **Mô hình CNN (Convolutional Neural Networks):** CNN (Convolutional Neural Networks) là một loại mô hình học máy tiên tiến lý tưởng để giải quyết vấn đề dữ liệu hình ảnh. Nó giúp chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao như hiện nay. CNN được sử dụng nhiều trong các bài toán nhận dạng các object trong ảnh. Để tìm hiểu tại sao thuật toán này được sử dụng rộng rãi cho việc nhận dạng (detection), chúng ta hãy cùng tìm hiểu về thuật toán này.
- **Mô hình Recurrent Neural Network (RNN):** RNN là một mô hình Deep Learning được sử dụng cho các bài toán xử lý dữ liệu tuần tự như ngôn ngữ tự nhiên và âm thanh. RNN sử dụng bộ nhớ để lưu trữ trạng thái thông tin từ những bước tính toán trước đó và sử dụng nó để dự đoán trạng thái tiếp theo.
- **Mô hình Long Short-Term Memory (LSTM):** LSTM (Long Short-Term Memory) là một loại mô hình RNN (Recurrent Neural Network) được thiết kế để giải quyết vấn đề giảm gradient khi huấn luyện mạng RNN thông thường. Mạng LSTM sử dụng các cấu trúc được gọi là "cổng" để điều chỉnh thông tin và kiểm soát dòng thông tin đi qua mạng. LSTM có ưu điểm chính là có khả năng học được các mối quan hệ phức tạp trong dữ liệu chuỗi thời gian, bởi vì nó có khả năng giữ lại thông tin trong bộ nhớ dài hạn. Điều này làm cho LSTM trở thành một trong những mô hình RNN phổ biến nhất cho các nhiệm vụ liên quan đến xử lý dữ liệu chuỗi thời gian, bao gồm nhận dạng giọng nói, dịch thuật máy tính, xử lý ngôn ngữ tự nhiên và phân loại tín hiệu.

CHƯƠNG 2. THIẾT KẾ VÀ CÀI ĐẶT GIẢI PHÁP

2.1. Kiến trúc tổng quát hệ thống

Đầu tiên, dữ liệu đầu vào của mô hình được đọc từ tập dữ liệu các tệp âm thanh, sau đó thực hiện xử lý dữ liệu bằng cách điều chỉnh các tệp âm thanh về cùng thời lượng/kích thước, tại đây tiến hành tăng cường dữ liệu và thực hiện gán nhãn, đồng thời đầu ra gồm nhãn và dữ liệu âm thanh đã được tăng cường cũng được lưu trữ lại với mục đích so sánh với kết quả dự đoán, đánh giá mô hình.



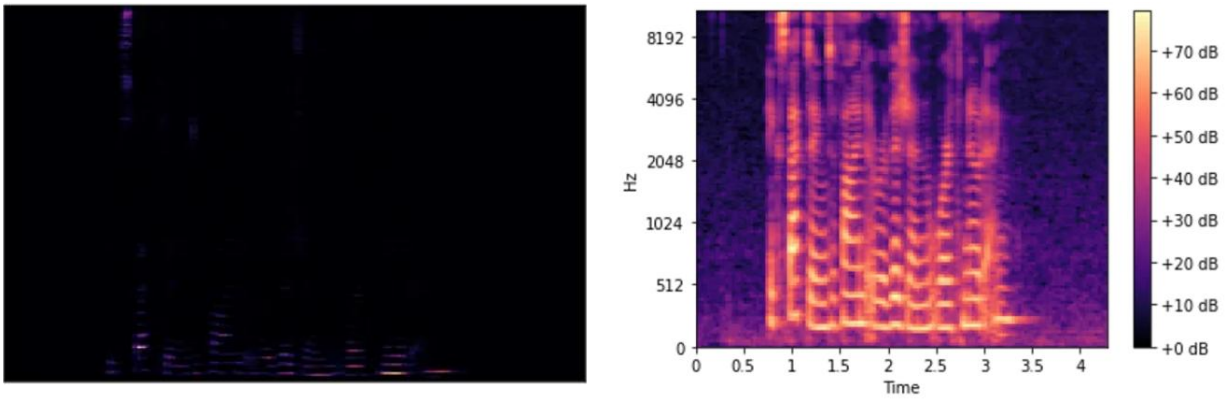
Hình 2.1. Kiến trúc tổng quan hệ thống.

Trong thời đại bùng nổ công nghệ thông tin và trí tuệ nhân tạo dữ liệu đóng vai trò quan trọng và không thể thiếu, đối với mục tiêu xây dựng mô hình đạt được kết quả tối thì một tập dữ liệu đủ tốt là điều vô cùng cần thiết. Để giải quyết vấn đề trên, người ta thường tìm cách thu thập thêm dữ liệu, tuy nhiên việc này sẽ không hề đơn giản đặc biệt khi dữ liệu sử dụng là âm thanh, để xử lý được chúng thường gây tốn kém thời gian, công sức và tiền bạc. Chính vì thế để đảm bảo có dữ liệu đáp ứng được mục đích phát triển mô hình chúng ta sẽ sử dụng các kỹ thuật cụ thể như điều chỉnh thời lượng/kích thước đoạn âm thanh, tăng cường dữ liệu âm thanh về cùng một tần số, chuyển tín hiệu âm thanh sang Mel spectrogram.

Phổ Mel chứa phép biến đổi Fourier ngắn hạn (STFT) cho mỗi khung của phổ (phổ năng lượng/biên độ), từ thang tần số tuyến tính sang thang đo Mel-logarit, sau đó đi qua bộ lọc để nhận được vector riêng (eigenvector). Các giá trị riêng này có thể được diễn giải một cách tương đối là sự phân bố năng lượng tín hiệu trên tần số thang Mel. Sau khi dữ liệu âm thanh được xử lý thành dữ liệu có độ dài 1-2 giây, tất cả dữ liệu sẽ được chuyển đổi thành Mel-spectrogram để có thể huấn luyện mạng nơ-ron tích chập (CNN) để nhận diện. Dữ liệu âm thanh thường có các đặc trưng phức tạp, vì vậy cần phải trích xuất những đặc trưng hữu ích để nhận diện âm thanh [1].

Mel-spectrogram là một trong những phương pháp hiệu quả để xử lý âm thanh và lấy mẫu ở tần số 16 kHz cho mỗi mẫu âm thanh. Trong thực nghiệm, mô hình sử dụng package Python có tên librosa để xử lý dữ liệu, với các thông số như sau: (n_fft = 1024, hop_length = 512, nmels = 128). Sau đó, tiếp tục gọi hàm power_to_db để chuyển đổi phổ năng lượng (bình phương biên độ) sang đơn vị decibel (dB) [1].

Trong **Hình 2.2**, là một ví dụ về Mel-spectrogram. Như có thể thấy từ hình, có một số khác biệt giữa các loại giọng nói khác nhau. Tuy nhiên, sau khi trộn lẫn nhiễu, một số chi tiết sẽ bị che khuất, điều này có thể giúp ta kiểm tra hiệu quả nhận diện giọng nói của mô hình trong thực tế. Đồng thời, cần phải trích xuất các đặc trưng của âm thanh và chuyển đổi chúng thành các hình ảnh đặc trưng, do đó có ba kênh giống như hình ảnh màu truyền thống.



Tín hiệu âm thanh ban đầu

Tín hiệu âm thanh khi chuyển đổi sang Mel spectrogram

Hình 2.2. Tín hiệu âm thanh biểu diễn ở dạng 1D và 2D

Tập dữ liệu sau khi xử lý được chia thành 2 phần với tỉ lệ 4 - 1 cho các quá trình huấn luyện (training) và đánh giá (testing) mô hình (training 4 - testing 1). Để đánh giá chính xác nhất hiệu suất của mô hình, nghiên cứu này thực hiện kỹ thuật đánh giá xác thực trên nhiều độ đo khác nhau (ACC, AUC, MCC, Training time) để chọn ra giải thuật tốt nhất (trong nghiên cứu này là CNN với dữ liệu âm thanh dưới dạng mảng 2D). Giải thuật được chọn sẽ được chạy thử nghiệm trên các bộ siêu tham số khách nhau để chọn ra mô hình với bộ siêu thanh số cho kết quả tốt nhất.

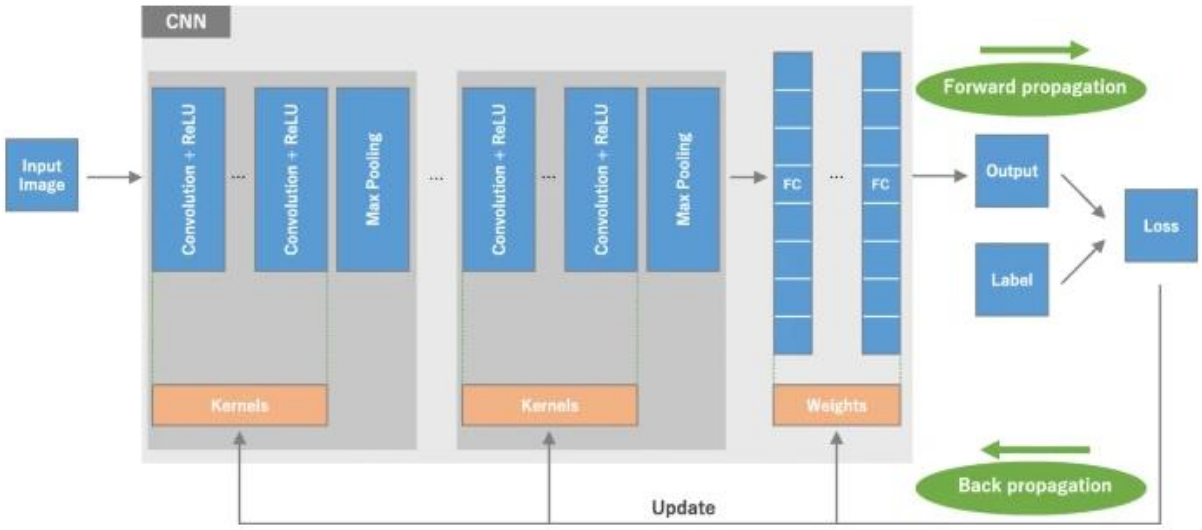
2.2. Xây dựng các mô hình

2.2.1. Kiến trúc CNN

CNN là một loại mô hình học sâu dùng để xử lý dữ liệu có dạng lưới, chẳng hạn như hình ảnh, được lấy cảm hứng từ cách tổ chức của vỏ não thị giác của động vật và được thiết kế để tự động và thích nghi học các cấp độ không gian của đặc trưng, từ mẫu đơn giản đến phức tạp. CNN là một cấu trúc toán học, thường bao gồm ba loại lớp (hoặc thành phần cơ bản): lớp tích chập (convolution), lớp giảm kích thước (pooling), và lớp kết nối đầy đủ (fully connected layer). Hai loại lớp đầu tiên, tích chập và giảm kích thước,

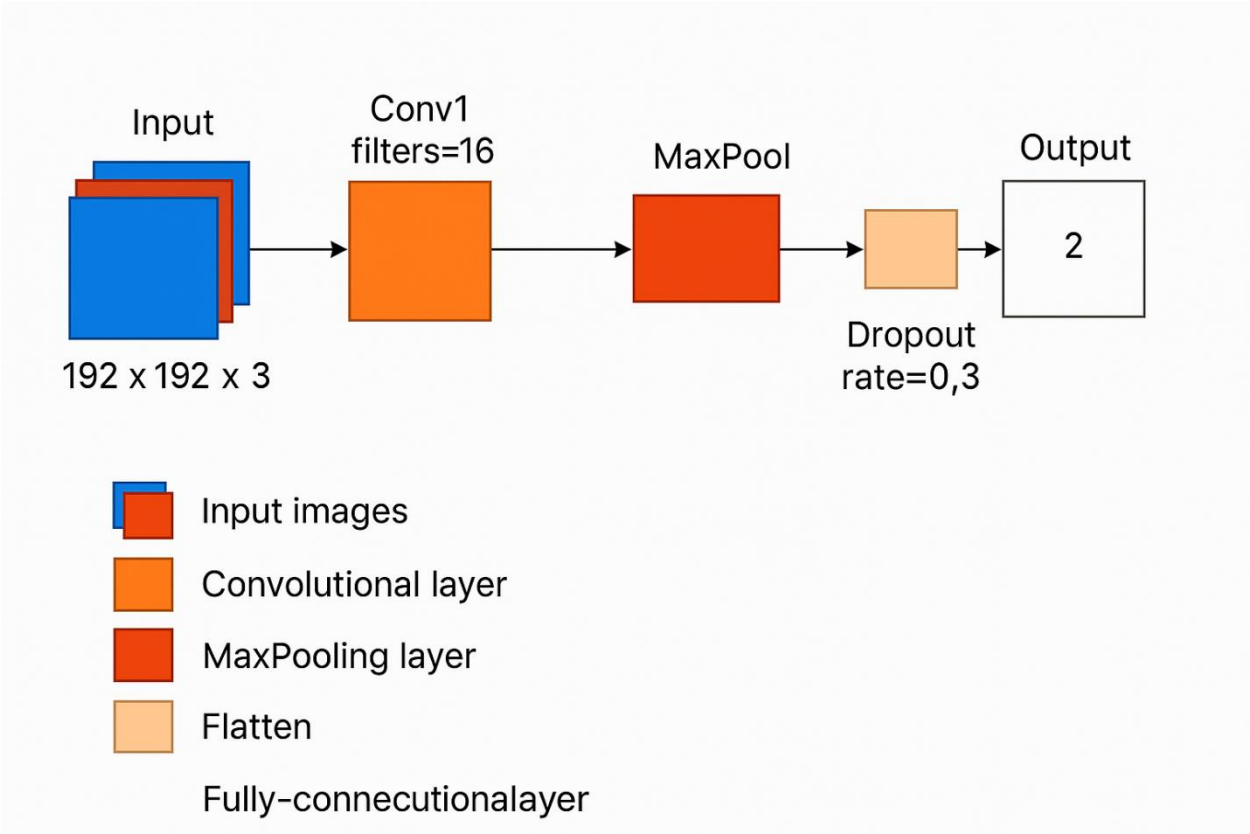
thực hiện trích xuất đặc trưng, trong khi lớp kết nối đầy đủ ánh xạ các đặc trưng đã trích xuất thành đầu ra cuối cùng, chẳng hạn như phân loại.

Lớp tích chập đóng vai trò quan trọng trong CNN, được cấu tạo từ nhiều phép toán toán học, như tích chập - một loại phép toán tuyến tính đặc biệt. Trong hình ảnh số, giá trị điểm ảnh được lưu trữ dưới dạng lưới hai chiều (2D), tức là một mảng các số, và một lưới nhỏ các tham số gọi là kernel - bộ trích xuất đặc trưng có thể tối ưu hóa - được áp dụng tại từng vị trí của hình ảnh, làm cho CNN cực kỳ hiệu quả trong xử lý hình ảnh, bởi vì đặc trưng có thể xuất hiện ở bất kỳ vị trí nào trên hình ảnh. Khi một lớp truyền đầu ra của nó cho lớp tiếp theo, các đặc trưng được trích xuất có thể trở nên phức tạp hơn theo cách phân cấp và tiến trình. Quá trình tối ưu hóa các tham số như kernel được gọi là huấn luyện, được thực hiện để giảm thiểu sự khác biệt giữa đầu ra và nhãn thực thông qua các thuật toán tối ưu hóa như lan truyền ngược (backpropagation) và hạ gradient (gradient descent), cùng với các phương pháp khác [1, 2].



Hình 2.1. Tổng quan về mô hình CNN

Tổng quan về kiến trúc của mạng nơ-ron tích chập (CNN) và quá trình huấn luyện. CNN được tạo thành từ việc xếp chồng nhiều thành phần cơ bản, bao gồm các lớp tích chập (convolution layers), các lớp giảm kích thước (pooling layers, ví dụ: max pooling), và các lớp kết nối đầy đủ (fully connected - FC layers). Hiệu suất của mô hình dựa trên các kernel và trọng số cụ thể được tính toán thông qua hàm mất mát (loss function) bằng lan truyền xuôi (forward propagation) trên tập dữ liệu huấn luyện. Các tham số có thể học được (learnable parameters), tức là kernel và trọng số, được cập nhật dựa trên giá trị của hàm mất mát thông qua lan truyền ngược (backpropagation) bằng thuật toán tối ưu hóa hạ gradient (gradient descent). ReLU, đơn vị tuyến tính được chỉnh sửa (rectified linear unit) [2].



Hình 2.2. Kiến trúc mạng nơ-ron tích chập (CNN) sử dụng trong nghiên cứu

Mô hình trên bao gồm các lớp (layers) cơ bản thường gặp trong mô hình học sâu xử lý dữ liệu hình ảnh.

- ***InputLayer***: đây là lớp đầu vào của mô hình. Nó định dạng dữ liệu đầu vào với kích thước dựa trên features[0].shape, tức là cấu trúc dữ liệu ban đầu.
- ***Conv2D(16, 3, padding='same', activation=keras.activations.relu)***: lớp tích chập (Convolutional Layer) thứ nhất với 16 bộ lọc (filters), kích thước bộ lọc là 3x3. Padding='same' đảm bảo kích thước của đầu ra không thay đổi so với đầu vào. Hàm kích hoạt là ReLU (Rectified Linear Unit), rất phổ biến để tạo tính phi tuyến trong mạng.
- ***MaxPooling2D(2)***: lớp giảm kích thước (Pooling Layer) với cửa sổ 2x2. Nó chọn giá trị lớn nhất trong mỗi vùng, giúp giảm số lượng tham số và làm nổi bật đặc trưng quan trọng.
- ***Flatten()***: lớp làm phẳng dữ liệu đầu vào từ dạng không gian 2D (từ ảnh hoặc đặc trưng tích chập) thành một vector 1D để sử dụng trong các lớp fully connected.
- ***Dropout(0.2)***: lớp Dropout ngẫu nhiên loại bỏ 20% các neuron trong quá trình huấn luyện, giúp giảm thiểu tình trạng quá khớp (overfitting).
- ***Dense(128, activation=keras.activations.relu)***: Lớp fully connected (kết nối đầy đủ) với 128 neuron và hàm kích hoạt ReLU. Đây là nơi xử lý các đặc trưng tổng quát.

- ***Dense(2, activation=keras.activations.sigmoid)***: Lớp đầu ra với 2 neuron, sử dụng hàm kích hoạt sigmoid. Điều này phù hợp với bài toán phân loại nhị phân hoặc đa nhãn (multi-label classification).

2.3. Giải pháp cài đặt

Nghiên cứu được xây dựng và phát triển trên nên nền tảng Google Colab, là nơi thực hiện huấn luyện tạo ra các mô hình chuẩn đoán, đánh giá kết quả sau quá trình huấn luyện. Trong hệ thống, các kiến trúc của mô hình được xây dựng trên ngôn ngữ lập trình Python, cho nên về cơ bản máy chủ của hệ thống cần cài đặt Python cùng các thư viện hỗ trợ cho việc xây dựng các mô hình máy học như: tensorflow của Google, keras, numpy, matplotplit,… Cùng với đó tập dữ liệu được sử dụng cho quá trình huấn luyện mô hình được lấy từ nền tảng Kaggle nên trong quá trình thiết lập dữ liệu chúng ta cần phải tải tệp dữ liệu trên về máy hoặc thực hiện các bước kết nối với nền tảng Kaggle.

Về phía client, để có thể sử dụng được nền tảng Google Colab, người dùng cần có một tài khoản Google và một trình duyệt web, có thể là: Google Chrome, Microsoft Edge, Firefox,… hoặc bất kỳ trình duyệt nào có thể kết nối đến địa chỉ của Google Colab.

CHƯƠNG 3. KIỂM THỬ VÀ ĐÁNH GIÁ

3.1. Kịch bản kiểm thử

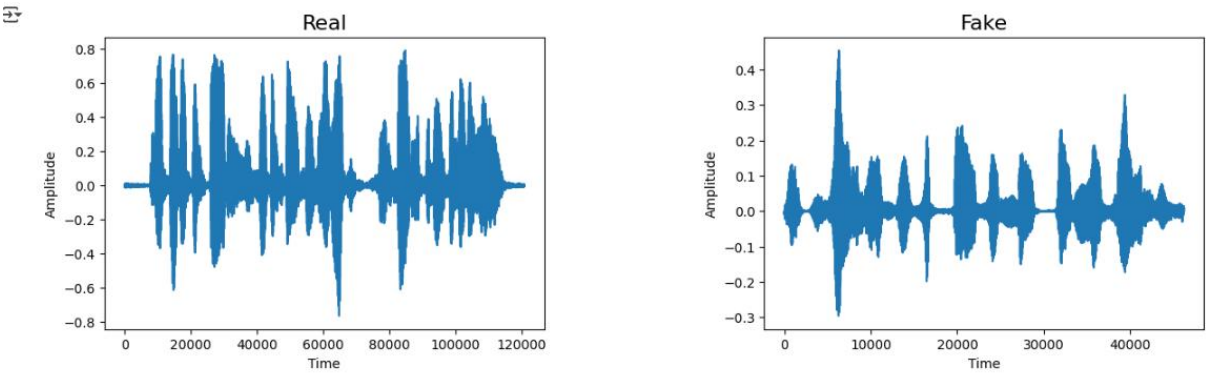
Mục tiêu của việc kiểm thử là đánh giá hiệu suất của mô hình máy học hỗ trợ nhận dạng giọng thật và giọng giả đồng thời kiểm tra độ chính xác của mô hình, qua đó đánh giá khách quan tính thực dụng và các đóng góp thực tế có thể đạt được của hệ thống.

3.1.1. Mô tả tập dữ liệu

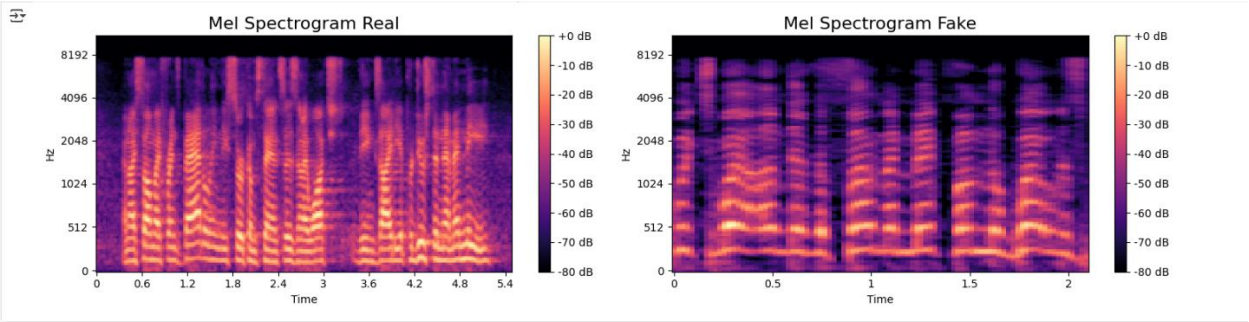
Trong quá trình tạo mô hình, việc nghiên cứu và huấn luyện mô hình được thực hiện trên tập dữ liệu Dataset Fake-or-Real (FoR) tập dữ liệu bao gồm câu nói từ cả giọng nói con người thật và giọng nói được máy tính tạo ra. Tập dữ liệu này được thiết kế để huấn luyện các bộ phân loại nhằm phát hiện giọng nói tổng hợp. Nó bao gồm dữ liệu từ các giải pháp tái tạo âm thanh tiên tiến như Deep Voice 3, Google WaveNet, và các nguồn giọng nói con người như Arctic Dataset, LJSpeech Dataset, VoxForge Dataset, cùng với các bản ghi âm tùy chỉnh [3]. Tất cả các dữ liệu âm thanh đều thuộc định dạng WAV, trong đó 500 đoạn âm thanh được thu từ giọng nói của con người, 500 đoạn âm thanh được thu từ giọng của các mô hình AI hiện đại tất cả đều có thời lượng khác nhau trung bình từ 2 giây đến 10 giây nên đòi hỏi việc điều chỉnh về cùng một kích thước/thời lượng khi đưa vào mô hình huấn luyện. Mẫu hình ảnh của các lớp được minh họa trong **Hình 3.1** và **Hình 3.2**.

Bảng 3.1. Số lượng samples ở mỗi lớp của tập dữ liệu Dataset Fake-or-Real

Class (Lớp)	Samples (Số lượng)
Fake (Giọng thu từ các mô hình AI)	500
Real (Giọng thu từ người nói)	500
Total (Tổng cộng)	1000



Hình 3.1. Biểu đồ minh họa âm thanh được biểu diễn theo biên độ và thời gian



Hình 3.2. Độ lớn âm thanh được biểu diễn theo thời gian và tần số

Trong nghiên cứu này, tập dữ liệu gốc được xử lý bằng các đưa tất cả các đoạn âm thanh về cùng kích thước/thời lượng 2 giây, đồng thời lưu trữ lại nhãn của lớp làm đầu ra cho mô hình phân lớp và masks cho mô hình phân vùng. Sau đó áp dụng kỹ thuật tăng cường dữ liệu và chia thành các tập dữ liệu phục vục cho việc huấn luyện và đánh giá xác thực chéo mô hình.

3.1.2. Môi trường thực nghiệm

Nghiên cứu được thực hiện trên nên tảng ngôn ngữ lập trình Python sử dụng các thư viện mã nguồn mở hỗ trợ cho các mô hình máy học như: Tensorflow của Google, cùng với Keras - một mã nguồn mở cho mạng nơ-ron hỗ trợ chạy mô hình trên cả CPU và GPU. Đề tài được thực nghiệm trên môi trường cụ thể quá trình xây dựng kiến trúc mô hình, đồng thời thực hiện đánh giá và chọn lọc mô hình tốt nhất được thực hiện trên Google Colab (Google Colaboratory), một sản phẩm của dự án Google Research với mục đính hỗ trợ chạy mã Python trực tiếp thông qua trình duyệt web. Chi tiết cấu hình môi trường phát triển được trình bày trong **Bảng 3.2**:

Bảng 3.2. Bảng chi tiết cấu hình môi trường phát triển.

Thành phần	**Cấu hình**
CPU	vCPUs
RAM	12.7 GB
Ổ cứng	107.7 GB
Trình duyệt	Microsoft Edge

3.1.3. Cơ sở đánh giá

Đề tài được xây dựng và đánh giá dựa trên đầu vào của dữ liệu âm thanh chia thành 2 loại: dữ liệu gốc được điều chỉnh về thời lượng tiêu chuẩn và dữ liệu tăng cường được điều chỉnh về thời lượng tiêu chuẩn. Cả 2 loại dữ liệu sẽ được đưa vào tập train (tập huấn luyện) và tập test (tập kiểm tra) của mô hình CNN, nghiên cứu thực hiện đánh giá trên 3 độ đo: ACC, AUC, MCC và Time training. Chi tiết về các độ đo được trình bày bên dưới.

- ***Accuracy (ACC):*** độ chính xác, là tỷ lệ giữa số mẫu dữ liệu (trong nghiên cứu này là số giọng) được dự đoán chính xác trên tổng số mẫu dữ liệu thực hiện dự đoán.

$$\boldsymbol{Accuracy = \frac{TN + TB + TM}{Total\ samples}}$$

Trong đó:

- True Normal (TN): số lượng hình ảnh thuộc lớp Normal được phân đúng vào lớp Normal.

- True Benign (TB): số lượng hình ảnh thuộc lớp Benign được phân đúng vào lớp Benign.

- True Malignant (TM): số lượng hình ảnh thuộc lớp Malignant được phân đúng vào lớp Malignant.

- Total samples: tổng số lượng hình ảnh kiểm thử.

- ***AUC (Area Under The Curve)***: AUC là một độ đo biểu diễn hiệu suất phân loại của mô hình, AUC có giá trị từ 0 đến 1, một mô hình dự đoán sai 100% có AUC bằng 0.0, một mô hình dự đoán đúng 100% có AUC bằng 1.0. AUC càng lớn, mô hình càng tốt.
- ***MCC (Matthews Correlation Coefficient):*** hệ số tương quan Matthews, dùng để đánh giá phẩm chất của một mô hình phân loại, được giới thiệu bởi Brian W. Matthews vào năm 1975 [4]. MCC là một hệ số tương quan giữa kết quả dự đoán của mô hình phân lớp và giá trị thực tế. MCC có giá trị dao động trong khoảng từ -1 đến +1, trường hợp MCC = +1 biểu thị cho mô hình phân loại hoàn hảo và chính xác tuyệt đối, MCC = 0 cho thấy mô hình vô dụng (tương tự như việc phán đoán ngẫu nhiên), MCC = -1 thể hiện việc mô hình dự đoán sai tuyệt đối so với thực tế.
- ***F1 - score:*** F1 - score là một thước đo thống kê được sử dụng rộng rãi trong học máy, đặc biệt trong các bài toán phân loại, để đánh giá hiệu quả của một mô hình. Giá trị này dao động từ 0 đến 1, trong đó 1 là mức lý tưởng, biểu thị mô hình hoạt động hoàn hảo. F1-score đặc biệt hữu ích trong trường hợp dữ liệu không cân đối, khi các lớp trong tập dữ liệu có sự chênh lệch đáng kể, và bạn cần cân nhắc giảm cả số lượng dương tính giả lẫn âm tính giả mà không ưu tiên một yếu tố nào hơn yếu tố còn lại [4]. Công thức tính F1 - score như sau:

$$\boldsymbol{\frac{2}{F1} = \frac{1}{Precision} + \frac{1}{Recall}}$$

Trong đó:

- Precision: độ chính xác và

- Recall: độ thu hồi

- ***Confusion Matrix:*** ma trận lỗi hay ma trân nhầm lẫn, là một phương pháp đo lường phổ biến và được sử dụng rộng rãi trong các bài toán phân lớp, confusion matrix

thường được biểu diễn dưới dạng bố cục bảng giúp cho việc hình dung hiệu suất của một mô hình phân lớp. Ma trận lỗi biểu diễn số lần xuất hiện của các lớp trong thực tế và số lần xuất hiện của các lớp được dự đoán. Một confusion matrix biểu diễn cho hiệu suất của mô hình phần lớp trong nghiên cứu này được biểu diễn tương tự như **Bảng 3.3**.

Bảng 3.3. Mẫu của một Confusion matrix

Actual Class			
Predicted Class	**Normal**	**Benign**	**Malignant**
Normal	TN	FB	FM
Benign	FN	TB	FM
Malignat	FN	FB	TM

Trong đó:

- True Normal (TN): số lượng hình ảnh thuộc lớp Normal được phân đúng vào lớp Normal.

- **True Benign (TB):** số lượng hình ảnh thuộc lớp Benign được phân đúng vào lớp Benign.

- **True Malignant (TM):** số lượng hình ảnh thuộc lớp Malignant được phân đúng vào lớp Malignant.

- **False Normal (FN):** số lượng hình ảnh không thuộc lớp Normal được phân vào lớp Normal.

- **False Benign (FB):** số lượng hình ảnh không thuộc lớp Benign được phân vào lớp Benign.

- **False Malignant (FM):** số lượng hình ảnh không thuộc lớp Malignant được phân vào lớp Malignant.

- ***Time training:*** thời gian để một mô hình thực hiện quá trình học sâu từ lúc đưa dữ liệu vào đến lúc mô hình kết thúc quá trình học sâu.

3.2. Kết quả kiểm thử

Để đánh giá hiệu suất của mô hình chuẩn đoán cho mỗi loại dữ liệu. Trong đó tập dữ liệu được xáo trộn và chia thành 5 phần, 4 phần được dùng cho mục đích huấn luyện mô hình (tương đương 800 tệp .wav), phần còn lại để đánh giá hiệu quả của mô hình (tương đương 200 tệp .wav). Quá trình này được lặp lại 10 lần (10 epochs). Phương pháp

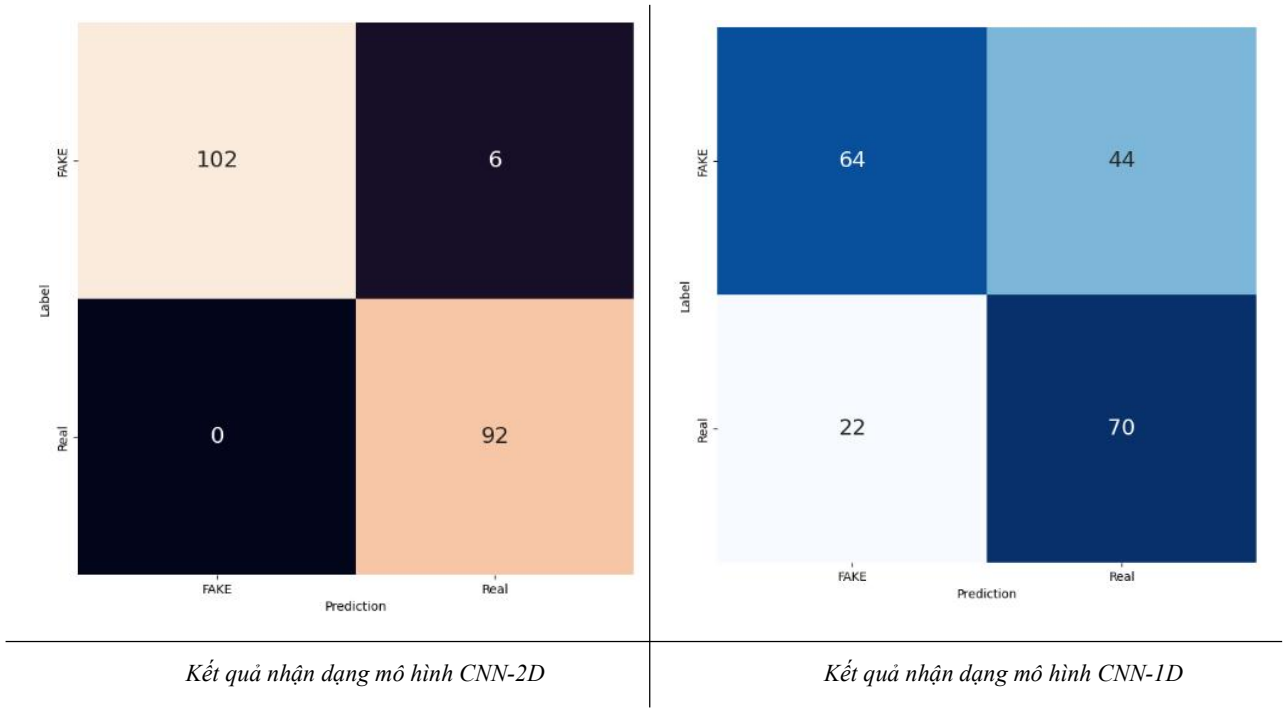
này nhằm đảm bảo tính chính xác của quá trình đánh giá mô hình trong điều kiện tập dữ liệu nhỏ và mất cân bằng giữa có các lớp.

Đối với nhiệm vụ phân lớp, mô hình CNN (Convolutional Neural Networks) được đào tạo và đánh giá qua Stratified5-Fold Cross-Validation 10 epochs với 2 loại tập dữ liệu: có áp dụng kỹ thuật tăng cường và không áp dụng kỹ thuật tăng cường (dữ liệu âm thanh gốc). Kết quả được tình bày trong **Bảng 3.4**.

Bảng 3.1. Kết quả đánh giá mô hình CNN ở hai loại dữ liệu.

Mô hình	Loại dữ liệu	ACC	AUC	MCC	F1-score	Training Time
CNN	2D	0.9765	0.9903	0.8999	0.9467	37.129 (giây)
	1D	0.67	0.7442	0.3558	0.6796	373.84 (giây)

Đầu tiên, so sánh trong tập dữ liệu huấn luyện không sử dụng tăng cường dữ liệu, ta có thể nhận thấy rõ mô hình CNN dữ liệu gốc (dữ liệu 1D) cho kết quả thấp hơn so với mô hình CNN sử dụng dữ liệu được tăng cường (dữ liệu 2D). Trong đó, mô hình CNN với dữ liệu gốc (dữ liệu 1D) cho độ chính xác thấp lần lượt với ACC = 0.740, AUC = 0.740, MCC = 0.4783, F1-score = 0.7574 so với mô hình CNN với dữ liệu được tăng cường (dữ liệu 2D) lần lượt với ACC = 0.9749, AUC = 0.9941, MCC = 0.911, F1-score = 0.9496. Bên cạnh đó thời gian huấn luyện của mô hình CNN sử dụng dữ liệu gốc (dữ liệu 1D) là 743,48 (giây) cao hơn so với thời gian huấn luyện của mô hình CNN sử dụng dữ liệu đã được tăng cường (dữ liệu 2D) là 46.924 (giây).



Kết quả nhận dạng mô hình CNN-2D | *Kết quả nhận dạng mô hình CNN-1D*

Hình 3.1. Ma trận nhầm lẫn với tập dữ liệu 2D và 1D.

Với kết quả của Confusion Matrix trên cho thấy, mô hình phân lớp thực sự hiểu quả trong việc dự đoán chính xác tất cả các lớp, không xuất hiện tình trạng "học vẹt". Tuy nhiên ở kết quả của mô hình CNN-1D các ô có màu khá sẫm, tương đương với việc xuất hiện nhiều trường hợp nhầm lẫn, đặc biệt là nhầm lẫn giữa lớp giọng giả với lớp giọng thật. Điều này cho thấy có thể có cơ số dữ liệu âm thanh thuộc lớp giọng giả có những đặc điểm tương tự giọng thật mà mô hình CNN-1D chưa thể nhận điện được. Điều này được cải thiện một cách hiệu quả hơn khi tăng cường dữ liệu âm thanh gốc sang Mel spectrogram. Tuy nhiên, nhìn chung ta có thể thấy mô hình đã hoàn thành tốt nhiệm vụ phân loại dữ liệu âm thanh ra các nhãn tương ứng.

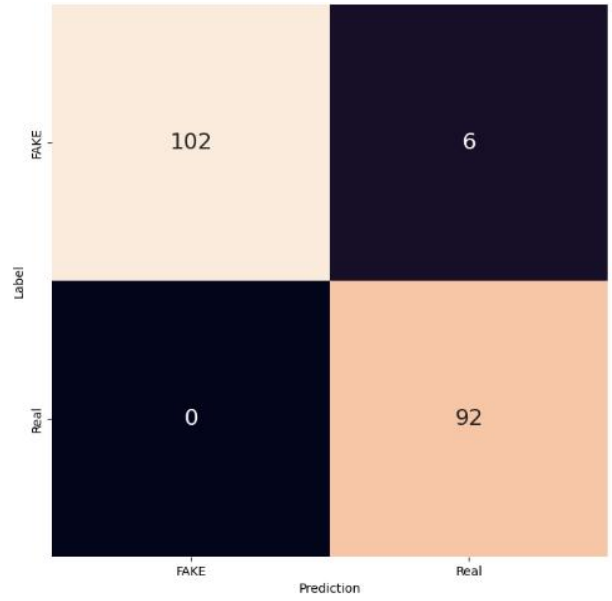
Nghiên cứu tiếp tục thực hiện thay đổi số lượng bộ lọc của lớp Convolutional trong mô hình CNN, với mục đích đánh giá sự tác động đến quá trình huấn luyện và hiệu suất của máy, kết quả cụ thể khi thay đổi số lượng bộ lọc (filters) ở lớp tích chập Convolutional Layer 1 và lớp tích chập Convolutional Layer 2 được biểu diễn trong **Bảng 3.5** dưới đây:

Bảng 3.2. Kết quả so sánh hiệu suất mô hình CNN khi tăng thêm layer.

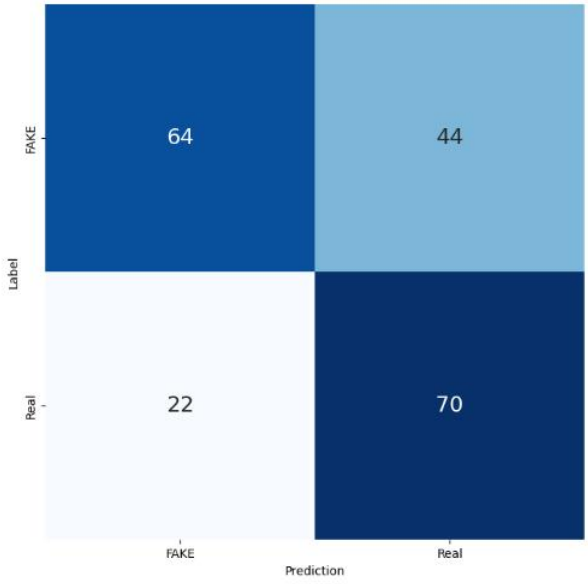
Mô hình	**Số lượng bộ lọc của lớp Convolutional**		**Loại dữ liệu**	**ACC**	**AUC**	**MCC**	**F1-score**	**Training Time**
	Layer 1	**Layer 2**						
CNN	16	0	2D	0.9765	0.9903	0.8999	0.9467	37.129 (giây)
	16	0	1D	0.67	0.7442	0.3558	0.6796	373.84 (giây)
CNN	16	32	2D	0.9749	0.9941	0.9110	0.9496	46.924 (giây)
	16	32	1D	0.740	0.778	0.4783	0.7574	743.48 (giây)

Bảng dữ liệu mô tả hiệu suất của các mô hình CNN với cấu hình tập trung vào số lượng bộ lọc trong lớp Convolutional và loại dữ liệu đầu vào (2D hoặc 1D). Kết quả cho thấy, mô hình CNN xử lý dữ liệu 2D vượt trội hơn so với dữ liệu 1D, khi độ chính xác (ACC), diện tích dưới đường cong ROC (AUC), hệ số Matthews (MCC), và F1-score đều cao hơn đáng kể. Hiệu suất của mô hình cũng bị ảnh hưởng bởi số lượng bộ lọc, trong đó mô hình với hai lớp tích chập và 32 bộ lọc ở lớp thứ hai cho kết quả tốt hơn so với mô hình chỉ sử dụng một lớp tích chập với 16 bộ lọc (từ 2% - 5%), đặc biệt ở các giá trị MCC và AUC trên dữ liệu 2D. Tuy nhiên, cấu hình phức tạp hơn cũng đòi hỏi thời gian

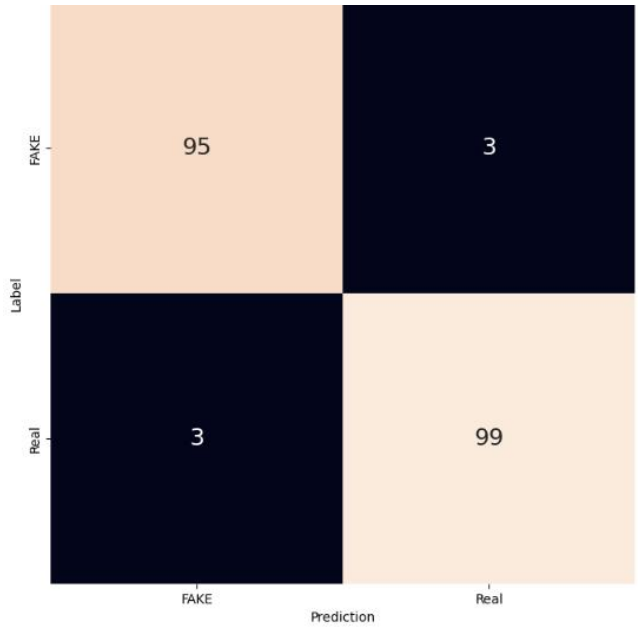
huấn luyện nhiều hơn. Qua kết quả trên có thể thấy mô hình CNN mang đến hiệu quả cao trong xử lý dữ liệu hình ảnh 2D và cấu hình 16/32 bộ lọc là lựa chọn tối ưu nếu cần hiệu suất cao. Trong trường hợp dữ liệu 1D hoặc hạn chế tài nguyên, mô hình với 16 bộ lọc vẫn là một phương án có thể đáp ứng bài toán mà ta đặt ra.



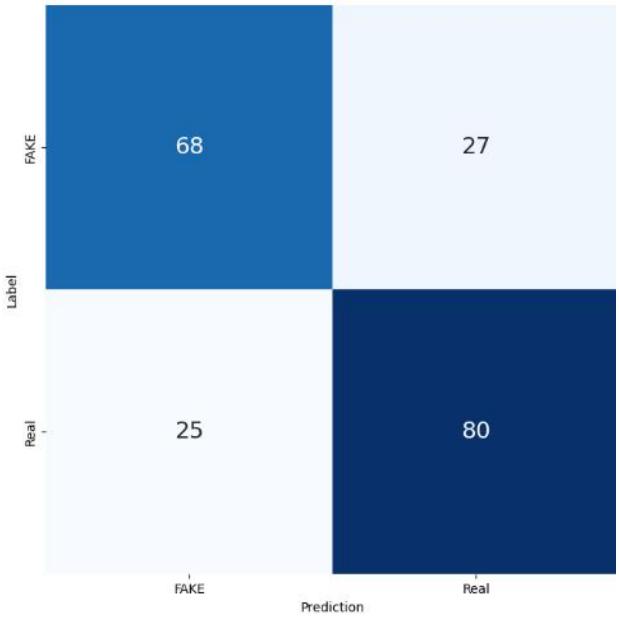
Kết quả nhận dạng mô hình CNN-2D với 1 lớp Converlutional



Kết quả nhận dạng mô hình CNN-1D với 1 lớp Converlutional



Kết quả nhận dạng mô hình CNN-2D với 2 lớp Converlutional



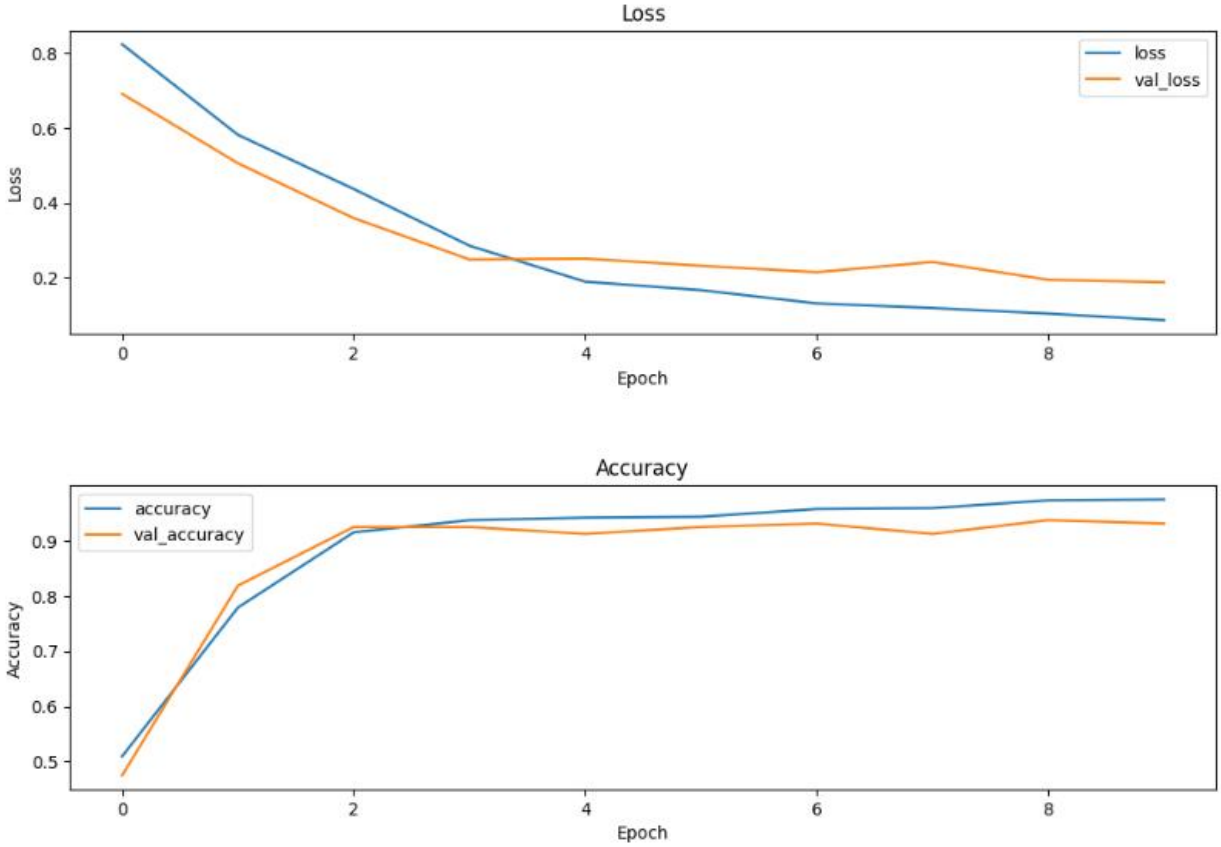
Kết quả nhận dạng mô hình CNN-1D với 2 lớp Converlutional

Hình 3.2. Ma trận nhầm lẫn ở các loại dữ liệu và số lượng lớp tích chập.

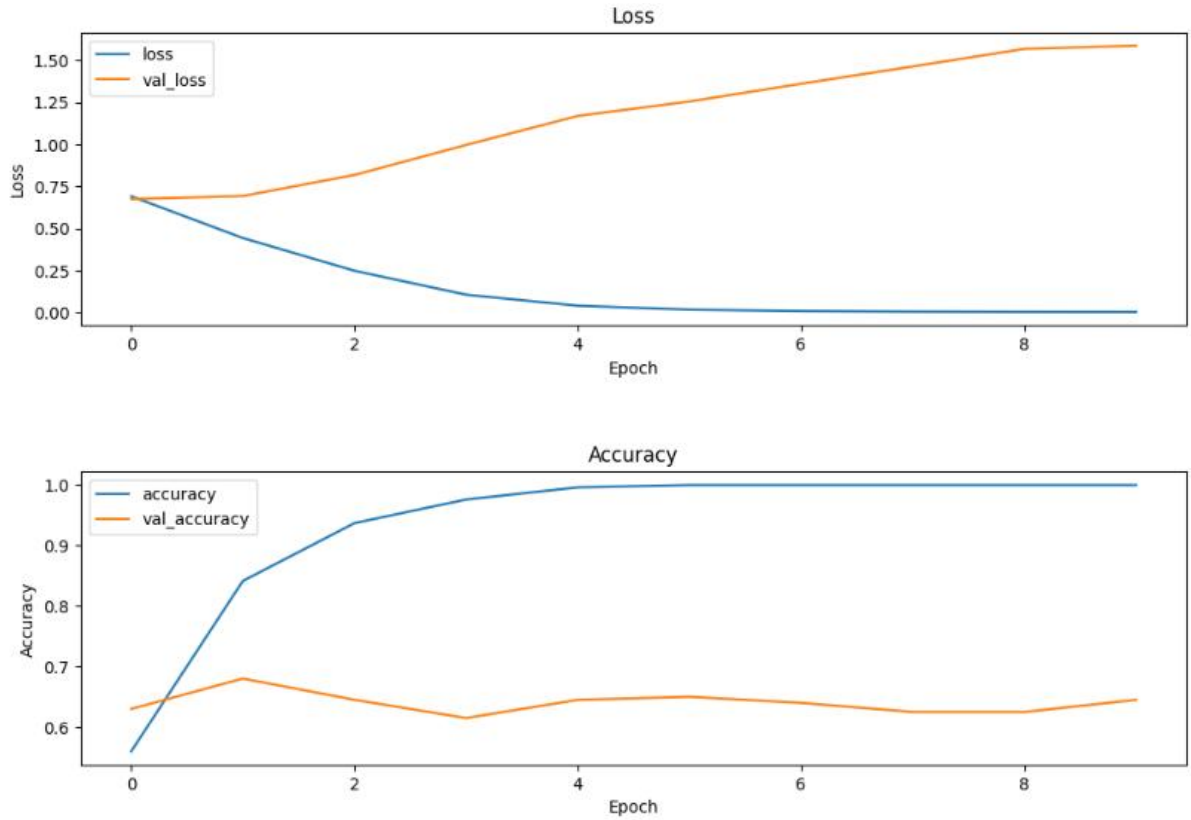
Dựa vào ma trận nhầm lẫn (Confusion Matrix) các mô hình CNN với dữ liệu đầu vào 2D cho thấy hiệu suất vượt trội hơn so với dữ liệu 1D trên cùng loại cấu trúc. Cụ thể,

mô hình CNN với một lớp tích chập (16 bộ lọc) đạt độ chính xác cao hơn đáng kể khi xử lý dữ liệu 2D, trong khi đó trên dữ liệu 1D, sai số trong nhận diện còn khá lớn. Đối với mô hình CNN với hai lớp tích chập (16/32 bộ lọc), dữ liệu 2D tiếp tục thể hiện ưu thế, đạt độ chính xác cao hơn và giảm đáng kể lỗi nhầm lẫn giữa giọng thật và giọng giả, trong khi dữ liệu 1D vẫn tồn tại mức độ sai lệch nhất định. Nhìn chung, nghiên cứu đã chứng minh rằng việc sử dụng dữ liệu 2D và cấu hình với hai lớp tích chập đem lại hiệu quả tốt hơn đáng kể trong nhiệm vụ nhận diện giọng. Điều này cho thấy tính ổn định của việc áp dụng CNN trong các bài toán nhận diện giọng nói với dữ liệu đầu vào 2D.

Trong nghiên cứu này còn cho thấy được độ chính xác và độ mất mát của từng mô hình cũng chịu ảnh hưởng từ loại dữ liệu đầu vào (trong nghiên cứu này là dữ liệu 1D và dữ liệu 2D) và số lượng lớp có trong mô hình được biểu diễn thông qua **Hình 3.3** và **Hình 3.4.**



Hình 3.3. Độ chính xác và độ mất mát của mô hình CNN-2D



Hình 3.4. Độ chính xác và độ mất mát của mô hình CNN-1D

Dựa trên các biểu đồ trong **Hình 3.3** và **Hình 3.4**, có thể quan sát thấy sự khác biệt rõ ràng trong quá trình huấn luyện của các mô hình. Trong **Hình 3.3**, train loss và validation loss đều giảm dần qua thời gian, điều này chứng tỏ mô hình học tốt và khả năng tổng quát hóa trên tập kiểm tra được duy trì ổn định. Ngược lại, trong **Hình 3.4**, mặc dù train loss tiếp tục giảm, validation loss lại tăng dần theo số lượng epochs, phản ánh hiện tượng overfitting khi mô hình học quá sâu vào tập huấn luyện nhưng không thể áp dụng hiệu quả trên dữ liệu mới.

Về độ chính xác (Accuracy), biểu đồ trong **Hình 3.3** cho thấy train accuracy và validation accuracy tăng dần đều và tiệm cận gần nhau, khẳng định mô hình hoạt động hiệu quả trên cả tập huấn luyện và tập kiểm thử. Tuy nhiên, trong **Hình 3.4**, train accuracy tăng mạnh nhưng validation accuracy lại dao động và có xu hướng giảm ở một số epochs, tiếp tục chỉ ra vấn đề overfitting.

Từ những kết quả này, có thể kết luận rằng mô hình sử dụng dữ liệu được tăng cường (CNN-2D) thể hiện hiệu suất ổn định và vượt trội hơn so với mô hình dùng dữ liệu âm thanh ban đầu (CNN-1D), đặc biệt trong việc duy trì sự cân bằng giữa train và validation.

CHƯƠNG 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

4.1. Kết luận

4.1.1. Kết quả đạt được

Sau quá trình thực hiện đề tài "***Sử dụng mô hình máy học để nhận diện giọng thật và giọng giả***" đã đạt được kết quả sau

- Nghiên cứu và xây dựng thành công mô hình máy học CNN sau đó thực hiện huấn luyện và đánh giá trên tập dữ liệu âm thanh đã chuẩn bị. Qua đó chọn ra được cách thức để chuyển đổi âm thanh đầu vào kết hợp với mô hình CNN với bộ siêu tham số phù hợp nhất cho kết quả nhận diện với độ chính xác cao và thời gian chạy ngắn.

4.1.2. Hạn chế

Mô hình phân loại giọng thật giọng giả đã cho kết quả tương đối tốt nhưng vẫn xảy ra hiện tượng overfitting (quá khớp) trong quá trình huấn luyện nên mô hình chưa thật sự tối ưu. Tập dữ liệu còn hạn chế dẫn đến quá trình học của mô hình chưa đủ nhiều để nhận diện được các dữ liệu âm thanh phức tạp hơn.

Phần tiền xử lý dữ liệu chưa thực sự tối ưu tốt, trong nhiều tình huống đặc biệt dữ liệu sau khi tiền xử lý không lấy ra được nội dung đáp ứng được mục tiêu dữ liệu cho việc huấn luyện mô hình dẫn đến mô hình còn nhận dạng nhầm lẫn giữa các tệp âm thanh có thời lượng và nội dung nói ít hoặc những tệp âm thanh có khoảng nghĩ kéo dài ở những giai đoạn đầu và cuối của đoạn âm thanh.

4.2. Hướng phát triển

Tăng cường thu thập dữ liệu từ nhiều nguồn khác nhau.

Chạy thực nghiệm mô hình hiện tại trên nhiều bộ tham số khác nhau. Nghiên cứu xây dựng và sử dụng các mô hình dữ liệu chuỗi thời gian (Time-series data) nổi tiếng như RNN, LSTM,… để cải thiện hiệu suất và kết quả chẩn đoán mô hình.

Nghiên cứu và tích hợp các giải thuật nhận diện dữ liệu âm thanh một cách chính xác để áp dụng trong giai đoạn tiền xử lý dữ liệu, tính toán khích thước hợp lý tránh tình trạng dữ liệu các đoạn âm thanh được dùng để huấn luyện mô hình không đáp ứng so với yêu cầu.

Phát triển các ứng dụng web/app để ứng dụng đề tài trong thực tế.

TÀI LIỆU THAM KHẢO

[1] Zhou, Q., Shan, J., Ding, W., Wang, C., Yuan, S., Sun, F., Li, H., & Fang, B. (2021). Cough Recognition Based on Mel-Spectrogram and Convolutional Neural Network. *Frontiers in Robotics and AI*, *8*, 580080. https://doi.org/10.3389/FROBT.2021.580080/BIBTEX

[2] Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, *9*(4), 611–629. https://doi.org/10.1007/S13244-018-0639-9/FIGURES/15

[3] *The Fake-or-Real (FoR) Dataset (deepfake audio)*. (n.d.). Retrieved April 9, 2025, from https://www.kaggle.com/datasets/mohammedabdeldayem/the-fake-or-real-dataset

[4] Chicco, D., & Jurman, G. (n.d.). *The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation*. https://doi.org/10.1186/s12864-019-6413-7