**TRƯỜNG ĐẠI HỌC QUY NHƠN**

KHOA CÔNG NGHỆ THÔNG TIN

-----


# BÁO CÁO TIỂU LUẬN

# HỌC PHẦN: PHÁT TRIỂN HỆ THỐNG TRÍ TUỆ NHÂN TẠO

**ĐỀ TÀI: Xây dựng hệ thống chuẩn đoán bệnh nám má**


| | |
|---|---|
| ***Giảng viên hướng dẫn:*** | TS. HỒ VĂN LÂM |
| ***Sinh viên thực hiện:*** | LÊ XUÂN TUYÊN |
| ***Mã sinh viên:*** | 4554100008 |
| ***Khoa:*** | CÔNG NGHỆ THÔNG TIN |
| ***Ngành và khóa:*** | TTNT 45 |
| ***Chuyên ngành:*** | TRÍ TUỆ NHÂN TẠO |

# LỜI CẢM ƠN

Trong suốt quá trình học tập và nghiên cứu thực hiện luận văn, ngoài nỗ lực của bản thân, em đã nhận được sự hướng dẫn nhiệt tình quý báu của quý Thầy Cô, cùng với sự động viên và ủng hộ của gia đình và bạn bè

Em xin chân thành cảm ơn Thầy TS Hồ Văn Lâm, người thầy kính mến đã hết lòng giúp đỡ, hướng dẫn, động viên, tạo điều kiện cho em trong suốt quá trình thực hiện và hoàn thành bài báo cáo.

Mặc dù đã có nhiều cố gắng, nỗ lực, nhưng do thời gian và kinh nghiệm nghiên cứu khoa học còn hạn chế nên không thể tránh khỏi những thiếu sót. Em rất mong nhận được sự góp ý của quý Thầy Cô cùng bạn bè để kiến thức của em ngày một hoàn thiện hơn.

Xin chân thành cảm ơn!

# MỤC LỤC

# Báo Cáo Phát Triển Hệ Thống AI: Hệ Thống Chuẩn Đoán Bệnh Nám Má

## Chương 1. Tổng quan báo cáo

### 1. Giới Thiệu

Bệnh nám má (melasma) là một rối loạn sắc tố da phổ biến, thường xuất hiện dưới dạng các mảng màu nâu hoặc xám nâu trên khuôn mặt, đặc biệt ở vùng má, trán, mũi, cằm hoặc môi trên. Tình trạng này ảnh hưởng chủ yếu đến phụ nữ, đặc biệt trong các giai đoạn thay đổi nội tiết tố như mang thai, sử dụng thuốc tránh thai, hoặc mãn kinh. Mặc dù không gây nguy hiểm nghiêm trọng đến sức khỏe, nám má có thể ảnh hưởng lớn đến thẩm mỹ và tâm lý của người mắc phải, dẫn đến giảm chất lượng cuộc sống.

Các yếu tố nguy cơ chính bao gồm tiếp xúc lâu dài với ánh nắng mặt trời (tia UV kích thích sản sinh melanin), thay đổi nội tiết tố (liên quan đến estrogen và progesterone), di truyền (tiền sử gia đình), và việc sử dụng mỹ phẩm không phù hợp (chứa hóa chất gây kích ứng hoặc tăng sắc tố). Việc chuẩn đoán sớm và chính xác đóng vai trò quan trọng trong việc kiểm soát bệnh, ngăn ngừa tình trạng nặng hơn, và áp dụng các biện pháp điều trị kịp thời như sử dụng kem chống nắng, thuốc bôi, hoặc liệu pháp laser.

Truyền thống, chuẩn đoán nám má dựa vào quan sát lâm sàng của bác sĩ da liễu, đôi khi kết hợp với đèn Wood để phân biệt với các bệnh da khác. Tuy nhiên, phương pháp này phụ thuộc nhiều vào kinh nghiệm của bác sĩ và không khả thi ở những khu vực thiếu chuyên gia hoặc cơ sở vật chất hạn chế. Vì vậy, việc phát triển một hệ thống trí tuệ nhân tạo (AI) hỗ trợ chuẩn đoán nám má không chỉ giúp tăng độ chính xác và hiệu quả mà còn mở rộng khả năng tiếp cận dịch vụ y tế cho bệnh nhân ở vùng sâu vùng xa.

Hệ thống này được xây dựng dưới dạng ứng dụng web sử dụng framework Flask, tích hợp mô hình học máy XGBoost để dự đoán nguy cơ mắc bệnh dựa trên dữ liệu khảo sát từ bệnh nhân. Ngoài ra, hệ thống còn cung cấp bài kiểm tra kiến thức để nâng cao nhận thức của người dùng về bệnh nám má, đồng thời hỗ trợ bác sĩ trong việc đưa ra quyết định lâm sàng.

### 1.2. Mục Tiêu

#### 1.2.1 Mục tiêu chính

Phát triển một hệ thống AI có khả năng dự đoán chính xác nguy cơ mắc bệnh nám má dựa trên các yếu tố nguy cơ như tuổi, tiền sử mang thai, sử dụng mỹ phẩm, mức độ tiếp xúc với ánh nắng mặt trời, và tiền sử gia đình.

#### 1.2.2 Mục tiêu phụ

Hỗ trợ bác sĩ da liễu trong việc đánh giá nguy cơ và đưa ra quyết định chuẩn đoán thông qua kết quả dự đoán và thông tin khảo sát.Nâng cao nhận thức của bệnh nhân về nguyên nhân, yếu tố nguy cơ, và cách phòng ngừa bệnh nám má thông qua bài kiểm tra kiến thức tích hợp.Thu thập dữ liệu thực tế từ bệnh nhân để cải thiện mô hình AI trong tương lai, đảm bảo tính cập nhật và chính xác.

### 1.3 Lĩnh vực

*Chuyên ngành*: Công nghệ phần mềm.
*Chuyên môn*: Lập trình web. Sử dụng ngôn ngữ HTML, CSS, PHP, MySQL, JavaScript để xây dựng trang web.
*Lĩnh vực liên quan*: Y tế

### 1.4 Yếu tố công nghệ

- Hệ điều hành Windows
- Phần mềm Visual Studio Code
- Website sẽ chạy được trên các trình duyệt web

### 1.5. Xác định các chức năng của phần mềm

**Yêu cầu chức năng**: Bệnh nhân cần nhập thông tin cá nhân (tuổi, số lần mang thai, tiền sử bệnh) và trả lời khảo sát kiến thức về bệnh rám má. Bác sĩ cần tra cứu thông tin bệnh nhân, xem kết quả dự đoán từ mô hình, và xác nhận chẩn đoán. Hệ thống phải tự động lưu trữ dữ liệu và cung cấp kết quả dự đoán.
**Yêu cầu phi chức năng**: Giao diện phải thân thiện, hỗ trợ tiếng Việt, và hoạt động tốt trên máy tính (responsive design). Bảo mật dữ liệu cá nhân bằng mã hóa và giới hạn quyền truy cập. Hiệu suất: Xử lý ít nhất 100 yêu cầu đồng thời mà không bị lỗi.

## Chương 2. Phân tích thiết kế

### 2.1 Xác định các tác nhân

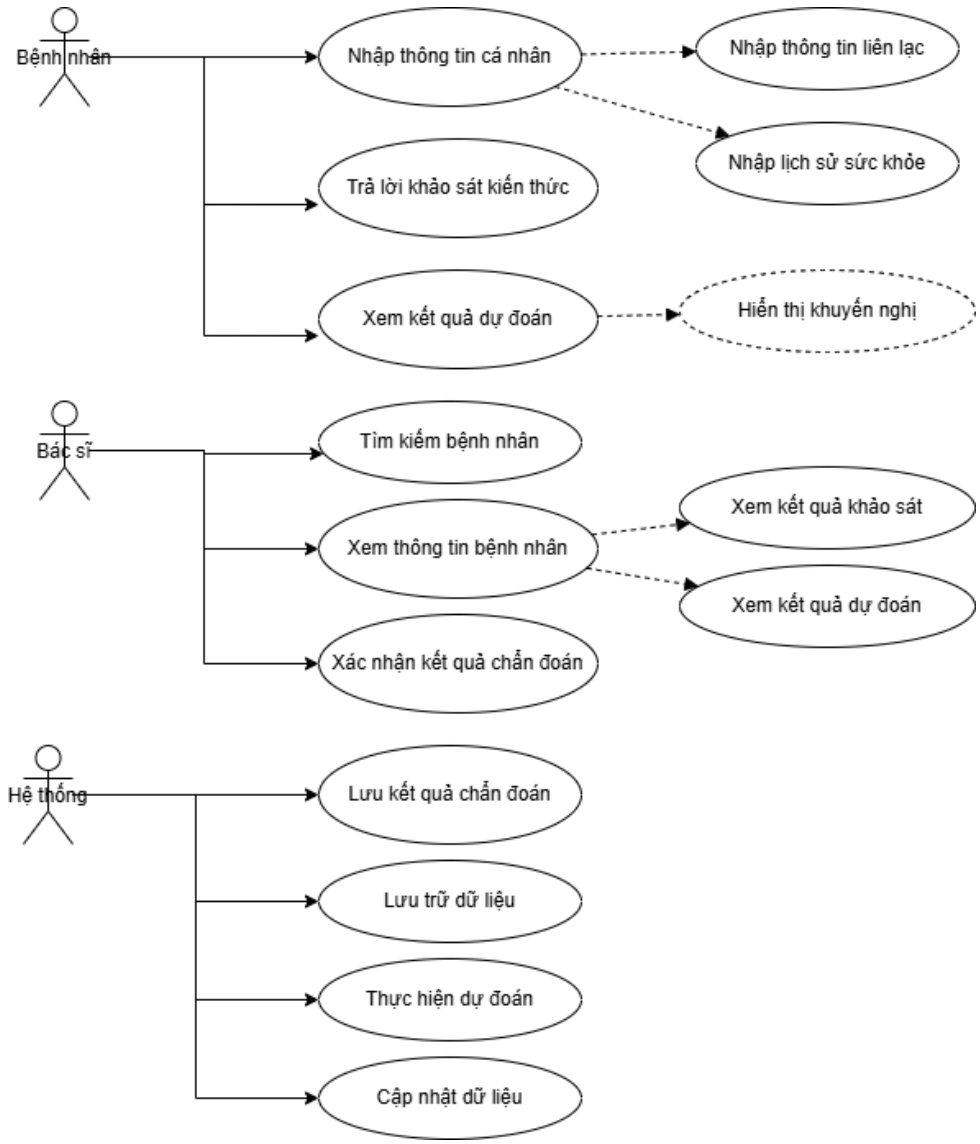
Actor là các thực thể tương tác trực tiếp với hệ thống. Dựa trên phân tích, các actor chính bao gồm:
-Bệnh nhân**:** Người cung cấp thông tin cá nhân, trả lời khảo sát, và xem kết quả dự đoán.
-Bác sĩ: Tra cứu thông tin bệnh nhân, xác nhận kết quả chẩn đoán, và lưu dữ liệu đã xử lý.
-Hệ thống: Tự động xử lý dữ liệu (lưu trữ, dự đoán, cập nhật) mà không cần can thiệp thủ công.

### 2.2 Biểu đồ usecase tổng quát

Biểu đồ usecase giúp xác định các trường hợp sử dụng chính và đảm bảo không bỏ sót chức năng quan trọng.

- Luồng chính:
  - Bệnh nhân: Nhập thông tin cá nhân → Trả lời khảo sát kiến thức → Xem kết quả dự đoán.
  - Bác sĩ: Tìm kiếm bệnh nhân → Xem thông tin bệnh nhân → Xác nhận kết quả chẩn đoán → Lưu kết quả chẩn đoán.
  - Hệ thống: Lưu trữ dữ liệu → Thực hiện dự đoán → Cập nhật dữ liệu.
- Mối quan hệ:
  - "Nhập thông tin cá nhân" bao gồm "Nhập thông tin liên lạc" và "Nhập lịch sử sức khỏe" (<<include>>).</include>
  - "Xem thông tin bệnh nhân" bao gồm "Xem kết quả khảo sát" và "Xem kết quả dự đoán" (<<include>>).</include>
  - "Xem kết quả dự đoán" có thể mở rộng để thêm "Hiển thị khuyến nghị" (<<extend>>).</extend>

Sơ đồ này giúp xác định các trường hợp sử dụng chính và đảm bảo không bỏ sót chức năng quan trọng.




## 2.3 Biểu đồ hoạt động

Biểu đồ hoạt động (Activity Diagram) mô tả luồng công việc hoặc quy trình trong hệ thống. Dưới đây là mô tả luồng hoạt động cho quy trình chẩn đoán bệnh:

- Luồng chính:
    1. **Bắt đầu**: Bệnh nhân truy cập hệ thống.
    2. **Nhập thông tin cá nhân**: Bệnh nhân điền dữ liệu (ví dụ: tuổi, tiền sử bệnh).
    3. **Trả lời khảo sát kiến thức**: Hoàn thành 18 câu hỏi.
    4. **Lưu dữ liệu**: Hệ thống lưu vào unsubmited_data.xlsx.
    5. **Thực hiện dự đoán**: Mô hình XGBoost xử lý và trả kết quả.
    6. **Xem kết quả**: Bệnh nhân xem kết quả ban đầu.
    7. **Tra cứu bởi bác sĩ**: Bác sĩ nhập ID bệnh nhân để xem thông tin.
    8. **Xác nhận chẩn đoán**: Bác sĩ kiểm tra và xác nhận (có/không).
    9. **Lưu kết quả**: Hệ thống cập nhật vào submitted_data.xlsx.
    10. **Kết thúc**: Quá trình hoàn tất.
- Quyết định:
    - Nếu bác sĩ không xác nhận, quay lại bước "Xem kết quả dự đoán" để điều chỉnh dữ liệu.

- Nếu có lỗi (ví dụ: dữ liệu thiếu), hệ thống thông báo và yêu cầu nhập lại.


# Chương 3: Triển  khai hệ thống

## 3.1 Dữ Liệu

### 3.1.1 Dữ liệu bệnh nhân

-Thông tin cá nhân: Bao gồm hơn 20 trường dữ liệu như tên, tuổi, địa chỉ, nghề nghiệp, trình độ học vấn, tình trạng hôn nhân, số lần mang thai, số con, tiền sử bệnh (bệnh tuyến giáp, bệnh cổ tử cung, v.v.), thời gian tiếp xúc với ánh nắng (giờ/ngày), tần suất sử dụng mỹ phẩm, và thói quen skincare.

-Phương pháp thu thập: Dữ liệu được nhập qua biểu mẫu trực tuyến tại patient_01_SurveyCollection.html, với các trường bắt buộc được đánh dấu để đảm bảo đầy đủ thông tin.

### 3.1.2 Dữ liệu khảo sát kiến thức

-Nội dung: Gồm 18 câu hỏi trắc nghiệm về bệnh nám má
-Phương pháp thu thập: Bệnh nhân trả lời qua patient_02_KnowlegdeTest.html, kết quả được lưu vào knowledgeTest_data.xlsx. Mỗi câu trả lời được mã hóa (0 cho sai, 1 cho đúng) để đánh giá mức độ hiểu biết.
-Ứng dụng: Dữ liệu này không chỉ giúp giáo dục bệnh nhân mà còn là một đặc trưng bổ sung cho mô hình dự đoán, vì kiến thức và thói quen có thể ảnh hưởng đến nguy cơ mắc bệnh.

### 3.1.3 Dữ liệu huấn luyện mô hình

-Tập dữ liệu lịch sử từ Benh_nhom1.csv, chứa thông tin của 700 bệnh nhân, bao gồm các đặc trưng như tuổi, số lần mang thai, tiền sử gia đình, mức độ rám má (nhẹ, trung bình, nặng), và nhãn (mắc bệnh: 1, không mắc: 0).

-Xử lý dữ liệu**:** Dữ liệu được làm sạch bằng cách loại bỏ giá trị thiếu, chuẩn hóa các biến số (ví dụ: tuổi từ 18-55 được chia thành các khoảng), và mã hóa các biến phân loại (ví dụ: nghề nghiệp: "nội trợ" = 0, "nhân viên văn phòng" = 1).

## 3.2. Mô Hình AI

### 3.2.1 Lựa Chọn Mô Hình

**Mô hình được chọn**: XGBoost Classifier với các siêu tham số ban đầu:

- max_depth=3: Độ sâu tối đa của cây quyết định.
- learning_rate=0.05: Tốc độ học.
- random_state=2: Đảm bảo khả năng tái tạo kết quả.
- n_estimators=40: Huấn luyện qua 40 epoch
- reg_lambda=1.0: Tham số lambda để kiểm soát độ phức tạp mô hình
- reg_alpha=0.5: Tham số alpha để kiểm soát độ phức tạp mô hình
- subsample=0.8: Sử dụng 80% mẫu ngẫu nhiên
- colsample_bytree=0.8: sử dụng 80% mẫu thử ngẫu nhiên

- **Lý do chọn XGBoost**:
  - Hiệu suất vượt trội trong các bài toán phân loại với dữ liệu có cấu trúc.
  - Khả năng xử lý tốt các đặc trưng tương tác phức tạp và dữ liệu không đồng đều.
  - So sánh với các mô hình khác:
  - **Random Forest**: Hiệu quả nhưng chậm hơn và ít linh hoạt trong việc tinh chỉnh. **Logistic Regression**: Đơn giản, nhưng không phù hợp với mối quan hệ phi tuyến giữa các đặc trưng.

### 3.2.2 Đặc Trưng Đầu Vào

Mô hình sử dụng các đặc trưng sau:

- Tuoi: Tuổi của bệnh nhân (số).

Một số nghiên cứu khoa học cho thấy: Tại các nước Đông Nam Á, nám má có thể bắt đầu xuất hiện từ 20-35 tuổi, tức là từ rất sớm so với các chủng tộc khác và cứ tăng 10 tuổi thì nguy cơ rám má tăng gấp 10,3 lần.

- So_Lan_Mang_Thai: Số lần mang thai (số).
- So_Lan_Sinh_Con: Số con đã sinh (số).
- Tien_Su_MangThai: Có/không xuất hiện nám khi mang thai (nhị phân).

  Có nghiên cứu khoa học cho thấy những phụ nữ có tiền sử nám má khi mang thai có cơ nguy nám má cao gấp 2,24 lần so với những phụ nữ không có tiền sử nám má khi mang thai. Bạn có tiền sử nám má khi mang thai vậy bạn hãy chú ý chăm sóc da dự phòng nám má từ sớm và điều trị ngay nếu thấy bắt đầu có biểu hiện nám má nhé!

- Tien_Su_Tranh_Thai: Có/không sử dụng thuốc tránh thai (nhị phân).

  Có nghiên cứu khoa học cho thấy phụ nữ dùng thuốc tránh thai sẽ có nguy cơ nám má gấp 2,15 lần so với những phụ nữ không dùng thuốc tránh thai.

- Tien_Su_Gia_Dinh: Có/không tiền sử gia đình mắc nám má (nhị phân).

  Có nghiên cứu khoa học cho thấy những phụ nữ có tiền sử gia đình rám má có nguy cơ rám má cao gấp 2,39 lần so với những phụ nữ không có tiền sử gia đình rám má. Bạn có người thân bị nám má sẽ dễ mắc bệnh nám má.

- Nghe_Nghiep: Nghề nghiệp (phân loại).
- Trinh_Do_Hoc_Van: Trình độ học vấn (phân loại).
- Tinh_Trang_hon_nhan: Tình trạng hôn nhân (phân loại).

  Khi kinh tế phát triển, nhu cầu thẩm mỹ ngày càng gia tăng.Những năm gần đây, trào lưu dùng mỹ phẩm dưỡng tẩy trắng da phát triển rầm rộ và đã xuất hiện nhiều hậu quả nghiêm trọng do sử dụng các sản phẩm bôi trắng da không an toàn như rám má và nhiều rối loạn sắc tố da khác, lão hóa da sớm, dị ứng mỹ phẩm. Tuy nhiên, vì hiệu quả điều trị còn hạn chế và chi phí điều trị rám má tốn kém nên chị em lại tiếp tục tìm cách bôi các sản phẩm tẩy trắng da không an toàn để giảm rám má, bất chấp hậu quả để lại

- Các đặc trưng bổ sung từ khảo sát: Thời gian tiếp xúc ánh nắng, loại mỹ phẩm.

  Có nghiên cứu khoa học cho thấy những phụ nữ có công việc phải tiếp xúc ánh nắng nhiều (> 1 giờ/ngày) có nguy cơ nám má cao gấp 1,69 lần so với những phụ nữ có thời gian tiếp xúc ánh nắng ít (≤ 1 giờ/ ngày). Có nghiên cứu khoa học cho thấy việc sử dụng mỹ phẩm dưỡng trắng không an toàn làm tăng nguy cơ rám má lên gấp 1,48 lần so với nhóm không dùng

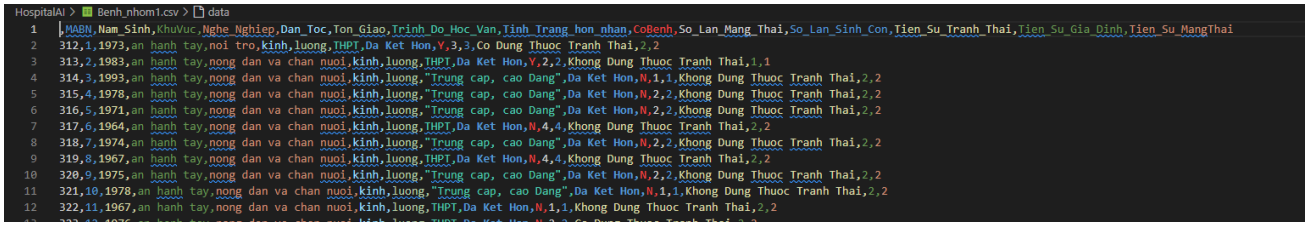
### 3.2.3 Tránh Overfitting

- **Regularization**: Sử dụng các tham số như lambda và alpha trong XGBoost để kiểm soát độ phức tạp của mô hình.
- **Early Stopping**: Dừng huấn luyện sớm nếu hiệu suất trên tập kiểm định không cải thiện sau 10 vòng lặp.
- **Cross-Validation**: Áp dụng 5-fold cross-validation để đảm bảo tính ổn định và giảm nguy cơ overfitting.

## 3.3. Quy Trình Phát Triển

### 3.3.1 Chuẩn bị dữ Liệu

Tập dữ liệu lịch sử từ Benh_nhom1.csv, chứa thông tin của 700 bệnh nhân, bao gồm các đặc trưng như tuổi, số lần mang thai, tiền sử gia đình, mức độ rám má (nhẹ, trung bình, nặng), và nhãn (mắc bệnh: 1, không mắc: 0).



```
HospitalAI > Benh_nhom1.csv > data
1  ,MABN,Nam_Sinh,KhuVuc,Nghe_Nghiep,Dan_Toc,Ton_Giao,Trinh_Do_Hoc_Van,Tinh_Trang_hon_nhan,CoBenh,So_Lan_Mang_Thai,So_Lan_Sinh_Con,Tien_Su_Tranh_Thai,Tien_Su_Gia_Dinh,Tien_Su_MangThai
2  312,1,1973,an hanh tay,noi tro,kinh,luong,THPT,Da Ket Hon,Y,3,3,Co Dung Thuoc Tranh Thai,2,2
3  313,2,1983,an hanh tay,nong dan va chan nuoi,kinh,luong,THPT,Da Ket Hon,Y,2,2,Khong Dung Thuoc Tranh Thai,1,1
4  314,3,1993,an hanh tay,nong dan va chan nuoi,kinh,luong,"Trung cap, cao Dang",Da Ket Hon,N,1,1,Khong Dung Thuoc Tranh Thai,2,2
5  315,4,1978,an hanh tay,nong dan va chan nuoi,kinh,luong,"Trung cap, cao Dang",Da Ket Hon,N,2,2,Khong Dung Thuoc Tranh Thai,2,2
6  316,5,1971,an hanh tay,nong dan va chan nuoi,kinh,luong,"Trung cap, cao Dang",Da Ket Hon,N,2,2,Khong Dung Thuoc Tranh Thai,2,2
7  317,6,1964,an hanh tay,nong dan va chan nuoi,kinh,luong,THPT,Da Ket Hon,N,4,4,Khong Dung Thuoc Tranh Thai,2,2
8  318,7,1974,an hanh tay,nong dan va chan nuoi,kinh,luong,"Trung cap, cao Dang",Da Ket Hon,N,2,2,Khong Dung Thuoc Tranh Thai,2,2
9  319,8,1967,an hanh tay,nong dan va chan nuoi,kinh,luong,THPT,Da Ket Hon,N,4,4,Khong Dung Thuoc Tranh Thai,2,2
10 320,9,1975,an hanh tay,nong dan va chan nuoi,kinh,luong,"Trung cap, cao Dang",Da Ket Hon,N,2,2,Khong Dung Thuoc Tranh Thai,2,2
11 321,10,1978,an hanh tay,nong dan va chan nuoi,kinh,luong,"Trung cap, cao Dang",Da Ket Hon,N,1,1,Khong Dung Thuoc Tranh Thai,2,2
12 322,11,1967,an hanh tay,nong dan va chan nuoi,kinh,luong,THPT,Da Ket Hon,N,1,1,Khong Dung Thuoc Tranh Thai,2,2
```

Chọn đặc trưng để huấn luyện mô hình

```python
# Select relevant columns
selected_columns = [
    'Tuoi', 'So_Lan_Mang_Thai', 'So_Lan_Sinh_Con',
    'Tien_Su_Tranh_Thai', 'Tien_Su_Gia_Dinh', 'Tien_Su_MangThai',
    'Nghe_Nghiep', 'Trinh_Do_Hoc_Van', 'Tinh_Trang_hon_nhan'
]
```

### 3.3.2 Tiền xử lý dữ liệu

Để đảm bảo dữ liệu sẵn sàng cho việc huấn luyện mô hình, các bước tiền xử lý sau được thực hiện:

**Đồng bộ tên cột**: Chuẩn hóa tên cột từ ứng dụng web và tệp huấn luyện (ví dụ: NAMSINH thành Nam_Sinh).

**Tính tuổi**: Sử dụng năm sinh và năm hiện tại (time.localtime().tm_year) để tính tuổi chính xác.

**Mã hóa biến phân loại**: Áp dụng LabelEncoder cho các cột như Nghe_Nghiep, Trinh_Do_Hoc_Van, Tinh_Trang_hon_nhan, Tien_Su_Tranh_Thai để chuyển đổi thành giá trị số.

**Chuẩn hóa biến số**: Sử dụng StandardScaler để chuẩn hóa các cột như Tuoi, So_Lan_Mang_Thai, So_Lan_Sinh_Con, đảm bảo phân phối dữ liệu đồng đều.

```python
# Encode categorical variables
categorical_cols = ['Nghe_Nghiep', 'Trinh_Do_Hoc_Van', 'Tinh_Trang_hon_nhan', 'Tien_Su_Tranh_Thai']
label_encoders = {}
for col in categorical_cols:
    le = LabelEncoder()
    df[col] = le.fit_transform(df[col])
    label_encoders[col] = le

# Scale numerical features
numerical_cols = ['Tuoi', 'So_Lan_Mang_Thai', 'So_Lan_Sinh_Con']
scaler = StandardScaler()
df[numerical_cols] = scaler.fit_transform(df[numerical_cols])
```

### 3.3.3 Chia Dữ Liệu

- **Tỷ lệ chia**:
  - Tập huấn luyện: 80% (636 mẫu).
  - Tập kiểm định: 10% (80 mẫu).
  - Tập kiểm tra: 10% (79 mẫu).
- **Phương pháp**: Chia ngẫu nhiên với train_test_split từ scikit-learn, đảm bảo phân phối nhãn cân bằng giữa các tập.

```
# Split data into training and validation sets
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.2, random_state=42)
```

### 3.3.4 Huấn Luyện Mô Hình

- Sử dụng XGBClassifier với các siêu tham số đã định

```
# Train XGBoost model
model = XGBClassifier(
    n_estimators=40,
    max_depth=3,
    learning_rate=0.05,
    reg_lambda=1.0,
    reg_alpha=0.5,
    subsample=0.8,
    colsample_bytree=0.8,
    random_state=42,
    eval_metric=["logloss", "error"]
)
```

- Lưu mô hình (model.pkl), scaler (scaler.pkl), và encoder (encoder.pkl) để tái sử dụng

```
# Save model using XGBoost's save_model
model.save_model('my_trained_model.model')

# Save preprocessors
with open('scaler.pkl', 'wb') as f:
    pickle.dump(scaler, f)
with open('encoders.pkl', 'wb') as f:
    pickle.dump(label_encoders, f)
```

### 3.3.5 Đánh Giá Mô Hình

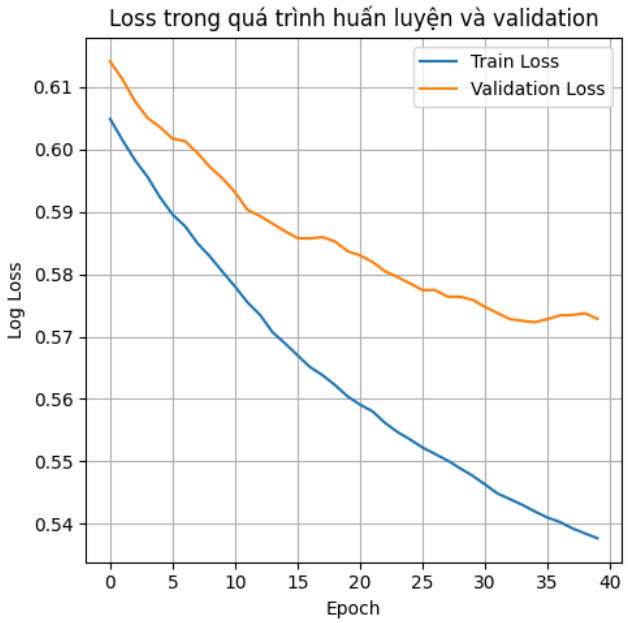
- **Chỉ số đánh giá**:
  - Accuracy: Tỷ lệ dự đoán đúng.
  - Log Loss: Mất mát trong quá trình huấn luyện

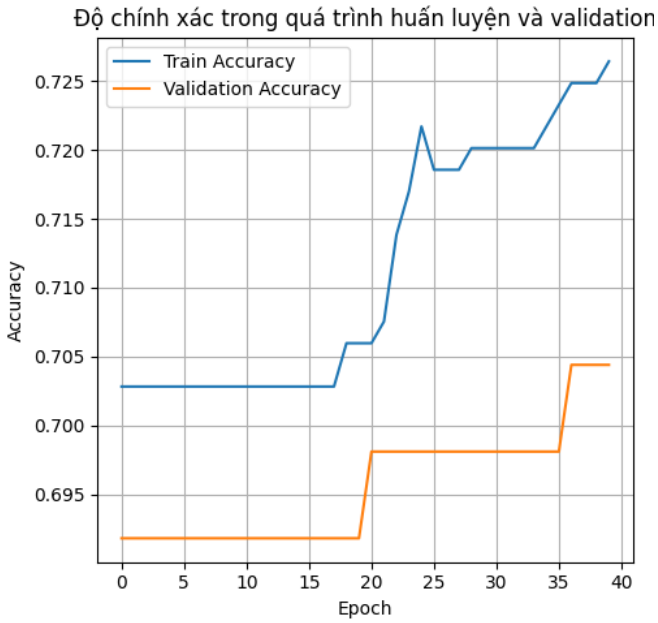
```python
# Extract evaluation results
results = model.evals_result()
epochs = range(len(results['validation_0']['logloss']))
train_loss = results['validation_0']['logloss']
val_loss = results['validation_1']['logloss']
train_error = results['validation_0']['error']
val_error = results['validation_1']['error']
train_accuracy = [1 - x for x in train_error]
val_accuracy = [1 - x for x in val_error]

# Calculate final accuracy on validation set
y_pred = model.predict(X_val)
final_accuracy = accuracy_score(y_val, y_pred)
final_log_loss = log_loss(y_val, model.predict_proba(X_val))
```

### 3.3.6 Kết quả huấn luyện

```
...  [0]   validation_0-logloss:0.60489   validation_0-error:0.29717   validation_1-logloss:0.61412   validation_1-error:0.30818
     [1]   validation_0-logloss:0.60146   validation_0-error:0.29717   validation_1-logloss:0.61122   validation_1-error:0.30818
     [2]   validation_0-logloss:0.59826   validation_0-error:0.29717   validation_1-logloss:0.60769   validation_1-error:0.30818
     [3]   validation_0-logloss:0.59559   validation_0-error:0.29717   validation_1-logloss:0.60504   validation_1-error:0.30818
     [4]   validation_0-logloss:0.59231   validation_0-error:0.29717   validation_1-logloss:0.60356   validation_1-error:0.30818
     [5]   validation_0-logloss:0.58957   validation_0-error:0.29717   validation_1-logloss:0.60173   validation_1-error:0.30818
     [6]   validation_0-logloss:0.58769   validation_0-error:0.29717   validation_1-logloss:0.60132   validation_1-error:0.30818
     [7]   validation_0-logloss:0.58495   validation_0-error:0.29717   validation_1-logloss:0.59943   validation_1-error:0.30818
     [8]   validation_0-logloss:0.58281   validation_0-error:0.29717   validation_1-logloss:0.59715   validation_1-error:0.30818
     [9]   validation_0-logloss:0.58037   validation_0-error:0.29717   validation_1-logloss:0.59535   validation_1-error:0.30818
     [10]  validation_0-logloss:0.57800   validation_0-error:0.29717   validation_1-logloss:0.59312   validation_1-error:0.30818
     [11]  validation_0-logloss:0.57547   validation_0-error:0.29717   validation_1-logloss:0.59032   validation_1-error:0.30818
     [12]  validation_0-logloss:0.57348   validation_0-error:0.29717   validation_1-logloss:0.58931   validation_1-error:0.30818
     [13]  validation_0-logloss:0.57070   validation_0-error:0.29717   validation_1-logloss:0.58810   validation_1-error:0.30818
     [14]  validation_0-logloss:0.56890   validation_0-error:0.29717   validation_1-logloss:0.58689   validation_1-error:0.30818
     [15]  validation_0-logloss:0.56700   validation_0-error:0.29717   validation_1-logloss:0.58578   validation_1-error:0.30818
     [16]  validation_0-logloss:0.56511   validation_0-error:0.29717   validation_1-logloss:0.58573   validation_1-error:0.30818
     [17]  validation_0-logloss:0.56381   validation_0-error:0.29717   validation_1-logloss:0.58596   validation_1-error:0.30818
     [18]  validation_0-logloss:0.56223   validation_0-error:0.29403   validation_1-logloss:0.58525   validation_1-error:0.30818
     [19]  validation_0-logloss:0.56042   validation_0-error:0.29403   validation_1-logloss:0.58370   validation_1-error:0.30818
     [20]  validation_0-logloss:0.55909   validation_0-error:0.29403   validation_1-logloss:0.58302   validation_1-error:0.30189
     [21]  validation_0-logloss:0.55802   validation_0-error:0.29245   validation_1-logloss:0.58195   validation_1-error:0.30189
     [22]  validation_0-logloss:0.55616   validation_0-error:0.28616   validation_1-logloss:0.58047   validation_1-error:0.30189
     [23]  validation_0-logloss:0.55470   validation_0-error:0.28302   validation_1-logloss:0.57955   validation_1-error:0.30189
     [24]  validation_0-logloss:0.55351   validation_0-error:0.27830   validation_1-logloss:0.57852   validation_1-error:0.30189
     ...
     [38]  validation_0-logloss:0.53845   validation_0-error:0.27516   validation_1-logloss:0.57373   validation_1-error:0.29560
     [39]  validation_0-logloss:0.53767   validation_0-error:0.27358   validation_1-logloss:0.57286   validation_1-error:0.29560
Độ chính xác: 0.7044
Log Loss: 0.5729
```



*Loss trong quá trình huấn luyện và validation*

Train Loss liên tục giảm, cho thấy mô hình học được tốt từ dữ liệu huấn luyện.Validation Loss cũng giảm nhưng chậm hơn, và từ khoảng epoch 20 trở đi, đường này gần như chững lại, có dấu hiệu bắt đầu dao động nhẹ.Điều này cho thấy mô hình không bị overfitting nghiêm trọng, nhưng cũng không còn cải thiện rõ rệt sau khoảng epoch 20–25.

*Độ chính xác trong quá trình huấn luyện và validation*

Train Accuracy tăng dần đều và vượt mốc 0.725 ở cuối quá trình huấn luyện. Validation Accuracy gần như không thay đổi trong suốt 20 epoch đầu, chỉ tăng nhẹ ở epoch cuối (lên gần 0.705). Cho thấy rằng mô hình đang học tốt trên tập huấn luyện nhưng lại khó cải thiện trên tập validation

## 3.4 Triển khai ứng dụng

Đưa mô hình đã huấn luyện vào API sẵn có : Open_source_AI APP

**model_creator.py**

Thêm xử lý pickle: Chế độ test tải self.scaler và self.label_encoders từ các file scaler.pkl và encoders.pkl bằng pickle.load. Có kiểm tra tính hợp lệ của label_encoders.

Thay đổi preprocess_data:
-Thêm ánh xạ cột (column_mapping) để đổi tên cột từ ứng dụng sang dữ liệu huấn luyện.
-Tính tuổi (Tuoi) dựa trên time.localtime().tm_year.
-Định nghĩa self.selected_columns để sử dụng trong huấn luyện và dự đoán.
-Mã hóa biến phân loại (categorical_cols) bằng LabelEncoder trong chế độ train, và xử lý giá trị chưa thấy trong chế độ test.
-Chuẩn hóa biến số (numerical_cols) bằng StandardScaler trong cả hai chế độ.
-Thêm nhãn (Label) từ cột RAMMA trong chế độ train.

Thay đổi train:
-Chia dữ liệu thành 3 tập (train, valid, test) với train_test_split.
-Lưu mô hình, scaler, và label_encoders vào file bằng pickle.dump.

**config.py**

Thêm đường dẫn mới: Thêm scaler_path và encoders_path để lưu trữ scaler.pkl và encoders.pkl.

Sửa đường dẫn: Sử dụng chuỗi thô (r"...") để tránh lỗi unicodeescape, r"C:\\Users\\XUAN TUYEN\\Desktop\\HospitalAI\\HospitalAI\\storages\\database\\unsubmited_data.xlsx".

Cập nhật training_features: Danh sách đặc trưng giờ đây bao gồm các cột như Age, Education_level, Job_1 đến Job_5, Economy_level, v.v., phản ánh dữ liệu từ form trong app.py, nhưng không được sử dụng trực tiếp trong model_creator.py hiện tại (vì model_creator.py sử dụng selected_columns).

**app.py**

Thay đổi /survey:
-Sửa lỗi cú pháp bằng cách thêm else để xử lý dữ liệu form khi area tồn tại.
-Thêm xử lý dữ liệu từ form (như number_of_pregnancies, Health_history_1 đến Health_history_4) với thử nghiệm lỗi (try-except) để tránh crash.

-Tạo patient_ID dựa trên tên và timestamp ngẫu nhiên, sau đó lưu dữ liệu vào unsubmited_data_path.

Thay đổi /knowledgeTest**:**
-Thêm logic tính điểm cho các câu trả lời (ví dụ: QA01 đến QA09, QB01 đến QB09) và lưu vào knowledgeTest_path.
-Truyền theory_results và practice_results sang /loadModel.

Thay đổi /loadModel:
-Lấy dữ liệu từ unsubmited_data_path dựa trên patient_ID, gọi ModelCreator để dự đoán, và cập nhật cột prediction.
-Thêm thông điệp (message) dựa trên số câu trả lời đúng.

Thay đổi /infoDisplay:
-Cập nhật cột RAMMA và DETAIL từ form, nhưng sửa lỗi append bằng cách sử dụng pd.concat hoặc gán trực tiếp (cần điều chỉnh thêm).
-Thêm các hàm như change_text, pregnant_change, v.v. để xử lý hiển thị dữ liệu.

### 3.5 Kết quả

Mô hình hoạt động tốt sau khi tích hợp vào API

Dưới đây là giao diện webapp sau khi tích hợp mô hình đã huấn luyện:

# THÔNG TIN CHẨN ĐOÁN

| | |
|---|---|
| Họ và tên * | Tuyên Lê |
| Ngày tháng năm sinh * | 2 / 1989 |
| Địa chỉ thường trú * | Số nhà, tên đường |
| Số điện thoại * | 094876xxx |
| Dân tộc * | Kinh |
| Nghề nghiệp hiện tại | Nông dân |
| Trình độ học vấn | Mù chữ, tiểu học |
| Tình trạng hôn nhân | Chưa kết hôn |
| Số lần mang thai | 1 |
| Số lần sinh con | 1 |
| Bạn có bị nám má khi mang thai không? | Không |
| Bạn có dùng thuốc tránh thai không? | Không |
| Gia đình bạn có ai đã hoặc đang bị nám má không? | Không |
| Bạn có mắc bệnh nội khoa không? | Có |
| Cụ thể hơn, bạn đã mắc bệnh gì? | ☐ Bệnh tuyến giáp<br>☐ Bệnh cổ tử cung<br>☐ Bệnh động kinh<br>☐ Bệnh khác |
| Một ngày bạn tiếp xúc với ánh nắng khoảng bao nhiêu giờ? | 5 |
| Bạn thường tiếp xúc với ánh nắng vào thời điểm nào trong ngày | ☑ Buổi sáng ☐ Buổi trưa ☐ Buổi chiều |
| Bạn có thường xuyên tiếp xúc với hóa chất không? | Có |
| Bạn thường tiếp xúc với loại hóa chất nào? | Xăng dầu |
| Bạn có sử dụng mỹ phẩm không? | Có |
| Bạn bắt đầu sử dụng mỹ phẩm từ khi nào? | 0 |
| Mỹ phẩm mà bạn hay sử dụng tên gì? | |
| Nước sản xuất loại mỹ phẩm mà bạn hay dùng | Việt Nam |
| Mỹ phẩm bạn thường mua ở đâu? | Cửa hàng tạp hóa |
| Mỹ phẩm bạn thường mua có giá tiền bao nhiêu? | Nếu giá tiền là 100 nghìn đồng, vui lòng điền là 100 |
| Mục đích của bạn sử dụng mỹ phẩm là gì? | ☐ Trắng da, mịn da,làm đẹp da<br>☐ Chữa nám má, tàn nhang<br>☐ Chống nhăn da<br>☐ Khác |

CHẨN ĐOÁN

# Câu hỏi khảo sát kiến thức, dự phòng rám má

Chọn câu trả lời cho các câu hỏi dưới đây.

## A. CÁC CÂU HỎI VỀ KIẾN THỨC ĐỐI VỚI BỆNH RÁM MÁ

| | | | |
|---|---|---|---|
| Rám má có phải là do di truyền không? | ○ Đúng | ○ Sai | ◉ Không biết |
| Rám má có phải là do nội tiết không? | ○ Đúng | ○ Sai | ◉ Không biết |
| Có phải rám má là do mang thai, sinh đẻ không? | ○ Đúng | ○ Sai | ◉ Không biết |
| Có phải rám má là do tiếp xúc với ánh nắng mặt trời hay không? | ○ Đúng | ○ Sai | ◉ Không biết |
| Có phải rám má là do dùng mỹ phẩm không đúng cách không? | ○ Đúng | ○ Sai | ◉ Không biết |
| Có phải rám má là do lão hóa da không? | ○ Đúng | ○ Sai | ◉ Không biết |
| Có phải rám má là do ảnh hưởng một số bệnh nội khoa không? | ○ Đúng | ○ Sai | ◉ Không biết |
| Có phải rám má có thể tự khỏi mà không cần điều trị không? | ○ Đúng | ○ Sai | ◉ Không biết |
| Có phải sử dụng kem chống nắng thường xuyên có thể phòng tránh rám má không? | ○ Đúng | ○ Sai | ◉ Không biết |

## B. CÁC CÂU HỎI THỰC HÀNH VỀ RÁM MÁ VÀ CÁC YẾU TỐ LIÊN QUAN

| | | |
|---|---|---|
| Bạn có Mang khẩu trang hàng ngày trước khi đi ra nắng không? | ○ Có | ◉ Không |
| Khẩu trang bạn sử dụng hàng ngày có trùm kín mặt, vải dày, sẫm màu không? | ○ Có | ◉ Không |
| Bạn có bôi kem chống nắng hàng ngày trước khi đi ra nắng 20-30 phút không? | ○ Có | ◉ Không |
| Bạn có sử dụng kem dưỡng trắng da ban đêm không? | ○ Có | ◉ Không |
| Bạn có được soi da, tư vấn sử dụng mỹ phẩm không? | ○ Có | ◉ Không |
| Mỹ phẩm bạn sử dụng có nhãn mác, nguồn gốc xuất xứ rõ ràng không? | ○ Có | ◉ Không |
| Mỹ phẩm bạn sử dụng có hạn sử dụng không? | ○ Có | ◉ Không |
| Bạn đã từng khám, tư vấn chăm sóc da ở cơ sở Y tế chưa? | ○ Có | ◉ Không |
| Bạn đã từng điều trị rám má ở cơ sở y tế chuyên khoa da liễu chưa? | ○ Có | ◉ Không |

NHẬN KẾT QUẢ ĐÁNH GIÁ


## Kết Luận

Hệ thống AI chuẩn đoán bệnh nám má là một công cụ hỗ trợ hiệu quả, tận dụng XGBoost để dự đoán nguy cơ dựa trên dữ liệu khảo sát. Với khả năng tích hợp vào quy trình y tế, hệ thống không chỉ hỗ trợ bác sĩ mà còn nâng cao nhận thức của bệnh nhân. Các cải tiến trong tương lai như phân tích hình ảnh và mở rộng dữ liệu sẽ giúp tăng độ chính xác và tính ứng dụng của hệ thống.