BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**



**ĐỒ ÁN**

**HỌC PHẦN MÁY HỌC ỨNG DỤNG**

**Đề tài**

**XÂY DỰNG MÔ HÌNH DỰ ĐOÁN QUYỀN TRUY CẬP TÀI NGUYÊN CỦA CÔNG TY AMAZON**

|  |  |
| --- | --- |
| **Nhóm sinh viên thực hiện:** | **Giảng viên hướng dẫn:** |
| Đổ Hiếu Nghĩa B2016985 | TS. Mã Trường Thành |
| Trần Công Nhật B2016989 |  |
| Diệp Nguyễn Minh Tuyến B2017016 |  |

Cần Thơ, 11/2023

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**



**ĐỒ ÁN**

**HỌC PHẦN MÁY HỌC ỨNG DỤNG**

**Đề tài**

**XÂY DỰNG MÔ HÌNH DỰ ĐOÁN QUYỀN TRUY CẬP TÀI NGUYÊN CỦA CÔNG TY AMAZON**

|  |  |
| --- | --- |
| **Nhóm sinh viên thực hiện:** | **Giảng viên hướng dẫn:** |
| Đổ Hiếu Nghĩa B2016985 | TS. Mã Trường Thành |
| Trần Công Nhật B2016989 |  |
| Diệp Nguyễn Minh Tuyến B2017016 |  |

Cần Thơ, 11/2023

**NHẬN XÉT CỦA GIẢNG VIÊN**

--------------------------------------------------------------------------------------------------------------------------------------------------------------------------

----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

*Cần Thơ, ngày tháng năm*

(Ký và ghi rõ họ tên)

# MỤC LỤC

[MỤC LỤC 1](#_Toc153046424)

[DANH MỤC HÌNH 3](#_Toc153046425)

[DANH MỤC BẢNG 4](#_Toc153046426)

[PHÂN CÔNG CÔNG VIỆC 5](#_Toc153046427)

[PHẦN NỘI DUNG 6](#_Toc153046428)

[CHƯƠNG I. TRỰC QUAN HÓA, XỬ LÝ TẬP DỮ LIỆU 6](#_Toc153046429)

[1. Tổng quan về dữ liệu 6](#_Toc153046430)

[1.1. Mô tả dữ liệu 6](#_Toc153046431)

[1.2. Ý nghĩa của dữ liệu 6](#_Toc153046432)

[2. Phân tích dữ liệu và tiền xử lý dữ liệu 7](#_Toc153046433)

[2.1. Phân tích dữ liệu 7](#_Toc153046434)

[2.2. Tiền xử lý dữ liệu 12](#_Toc153046435)

[3. Cấu hình máy tính huấn luyện mô hình 15](#_Toc153046436)

[CHƯƠNG II. HUẤN LUYỆN MÔ HÌNH 16](#_Toc153046437)

[1. Giải thuật KNN 16](#_Toc153046438)

[1.1. Giới thiệu 16](#_Toc153046439)

[1.2. Cách hoạt động của giải thuật 16](#_Toc153046440)

[1.3. Ưu điểm, nhược điểm của giải thuật 16](#_Toc153046441)

[1.4. Kết quả huấn luyện mô hình 17](#_Toc153046442)

[2. Giải thuật Naïve Bayes 17](#_Toc153046443)

[2.1. Giới thiệu 17](#_Toc153046444)

[2.2. Cách hoạt động của giải thuật 17](#_Toc153046445)

[2.3. Ưu điểm, nhược điểm của giải thuật 18](#_Toc153046446)

[2.4. Kết quả huấn luyện mô hình 18](#_Toc153046447)

[3. Giải thuật Decision Tree 18](#_Toc153046448)

[3.1. Giới thiệu 18](#_Toc153046449)

[3.2. Cách hoạt động của giải thuật 19](#_Toc153046450)

[3.3. Ưu điểm, nhược điểm của giải thuật 19](#_Toc153046451)

[3.4. Kết quả huấn luyện mô hình 19](#_Toc153046452)

[4. Giải thuật Random Forest 20](#_Toc153046453)

[4.1. Giới thiệu 20](#_Toc153046454)

[4.2. Cách hoạt động của giải thuật 20](#_Toc153046455)

[4.3. Ưu điểm, nhược điểm của giải thuật 21](#_Toc153046456)

[4.4. Kết quả huấn luyện mô hình 21](#_Toc153046457)

[CHƯƠNG III. ĐÁNH GIÁ VÀ TRIỂN KHAI MÔ HÌNH 23](#_Toc153046458)

[1. Đánh giá mô hình phân lớp 23](#_Toc153046459)

[2. Nhận xét kết quả thực nghiệm 24](#_Toc153046460)

[3. Triển khai mô hình 24](#_Toc153046461)

[PHẦN KẾT LUẬN 27](#_Toc153046462)

[1. Kết quả đạt được 27](#_Toc153046463)

[2. Hướng phát triển 27](#_Toc153046464)

[TÀI LIỆU THAM KHẢO 28](#_Toc153046465)

# DANH MỤC HÌNH

[Hình 1: Dataset Amazon employee access 6](file:///E:\maubaocao.docx#_Toc153045608)

[Hình 2: Histogram của thuộc tính RESOURCE 7](file:///E:\maubaocao.docx#_Toc153045609)

[Hình 3: Histogram của thuộc tính MGR\_ID 8](file:///E:\maubaocao.docx#_Toc153045610)

[Hình 4: Histogram của thuộc tính ROLE\_ROLLUP\_1 8](file:///E:\maubaocao.docx#_Toc153045611)

[Hình 5: Histogram của thuộc tính ROLE\_ROLLUP\_2 9](file:///E:\maubaocao.docx#_Toc153045612)

[Hình 6: Histogram của thuộc tính ROLE\_DEPTNAME 9](file:///E:\maubaocao.docx#_Toc153045613)

[Hình 7: Histogram của thuộc tính ROLE\_TITLE 10](file:///E:\maubaocao.docx#_Toc153045614)

[Hình 8: Histogram của thuộc tính ROLE\_FAMILY\_DESC 10](file:///E:\maubaocao.docx#_Toc153045615)

[Hình 9: Histogram của thuộc tính ROLE\_FAMILY 11](file:///E:\maubaocao.docx#_Toc153045616)

[Hình 10: Histogram của thuộc tính ROLE\_CODE 11](file:///E:\maubaocao.docx#_Toc153045617)

[Hình 11: kết quả tóm tắt thông tin về dataframe 12](file:///E:\maubaocao.docx#_Toc153045618)

[Hình 12: kết quả dataframe sau khi xóa cột id 13](file:///E:\maubaocao.docx#_Toc153045619)

[Hình 13: kết quả kiểm tra các mẫu có bị trùng lặp không? 13](file:///E:\maubaocao.docx#_Toc153045620)

[Hình 14: kết quả dataframe sau khi cân bằng lại dữ liệu 14](file:///E:\maubaocao.docx#_Toc153045621)

[Hình 15: dataset sau khi chuẩn hóa dữ liệu 14](file:///E:\maubaocao.docx#_Toc153045622)

[Hình 16: Kết quả độ chính xác của từng mô hình theo nghi thức đánh giá hold-out 23](file:///E:\maubaocao.docx#_Toc153045623)

[Hình 17: Kết quả độ chính xác của từng mô hình theo nghi thức đánh giá 15-fold 23](file:///E:\maubaocao.docx#_Toc153045624)

[Hình 18: Ma trận nhầm lẫn mô hình Random Forest 24](file:///E:\maubaocao.docx#_Toc153045625)

[Hình 19: Kết quả kiểm thử mô hình đã được triển khai 26](file:///E:\maubaocao.docx#_Toc153045626)

# DANH MỤC BẢNG

[Bảng 1: Bảng phân công công việc 5](#_Toc153045869)

[Bảng 2: Bảng cấu hình máy tính triển khai mô hình 15](#_Toc153045870)

[Bảng 3: Bảng kết quả huấn luyện mô hình KNN nghi thức hold-out 17](#_Toc153045871)

[Bảng 4: Bảng kết quả huấn luyện mô hình KNN nghi thức 15-fold 17](#_Toc153045872)

[Bảng 5: Bảng kết quả huấn luyện mô hình Naive Bayes nghi thức hold-out 18](#_Toc153045873)

[Bảng 6: Bảng kết quả huấn luyện mô hình Naive Bayes nghi thức 15-fold 18](#_Toc153045874)

[Bảng 7: Bảng kết quả huấn luyện mô hình Decision Tree nghi thức Hold-out 19](#_Toc153045875)

[Bảng 8: Bảng kết quả huấn luyện mô hình Decision Tree nghi thức 15-fold 20](#_Toc153045876)

[Bảng 9: Bảng kết quả huấn luyện mô hình Random Forest nghi thức hold-out 21](#_Toc153045877)

[Bảng 10: Bảng kết quả huán luyện mô hình Random Forest nghi thức 15-fold 22](#_Toc153045878)

[Bảng 11: Bảng test case kiểm thử ứng dụng 25](#_Toc153045879)

# PHÂN CÔNG CÔNG VIỆC

Bảng 1: Bảng phân công công việc

|  |  |  |
| --- | --- | --- |
| **STT** | **Họ và tên** | **Công việc** |
| 1 | Đổ Hiếu Nghĩa  (Nhóm trưởng) | 1. Thu thập dữ liệu 2. Đọc hiểu dữ liệu 3. Tiền xử lý dữ liệu 4. Huấn luyện mô hình (Nghi thức Hold out, KNN, Naïve Bayes, Decision Tree, Random Forest) 5. Triển khai mô hình thành ứng dụng 6. Soạn slide báo cáo |
| 2 | Trần Công Nhật | 1. Thu thập dữ liệu 2. Đọc hiểu dữ liệu 3. Huấn luyện mô hình (Nghi thức K-fold Decision Tree, Random Forest) 4. Triển khai mô hình thành ứng dụng |
| 3 | Diệp Nguyễn Minh Tuyến  (Thư ký) | 1. Thu thập dữ liệu 2. Đọc hiểu dữ liệu 3. Huấn luyện mô hình (Nghi thức K-fold KNN, Naïve Bayes) 4. Tổng hợp kết quả đánh giá mô hình 5. Viết bài báo cáo |

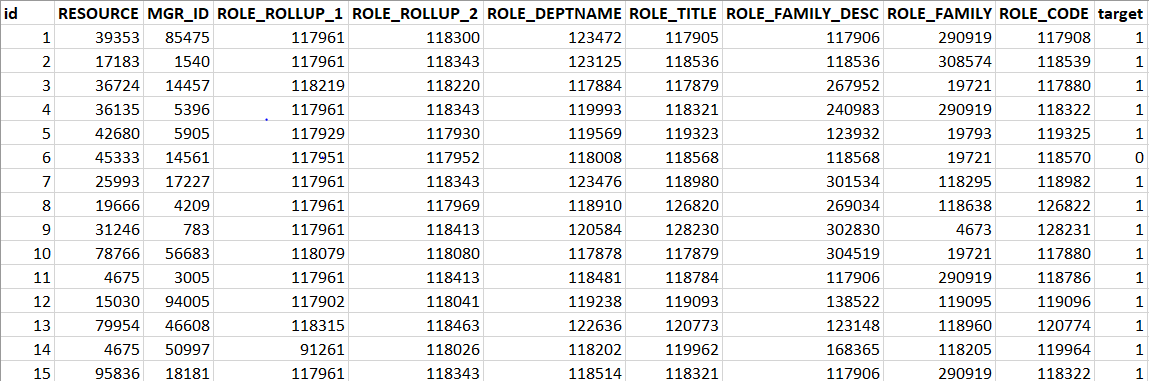
# PHẦN NỘI DUNG

# CHƯƠNG I. TRỰC QUAN HÓA, XỬ LÝ TẬP DỮ LIỆU

## 1. Tổng quan về dữ liệu

## 1.1. Mô tả dữ liệu

* Dữ liệu có tên: Amazon\_employee\_access
* Nguồn: OpenML ID 4135

 Mô tả: Dữ liệu bao gồm 32769 hàng và 11 cột, bộ dữ liệu bao gồm dữ liệu lịch sử thực được thu thập từ năm 2010 và 2011. Nhân viên được cho phép hoặc từ chối quyền truy cập vào tài nguyên theo thời gian. Dữ liệu được sử dụng để tạo ra một thuật toán có khả năng học hỏi từ dữ liệu lịch sử này để dự đoán sự chấp thuận/từ chối cho một nhóm nhân viên chưa từng thấy.

Hình 1: Dataset Amazon employee access

## 1.2. Ý nghĩa của dữ liệu

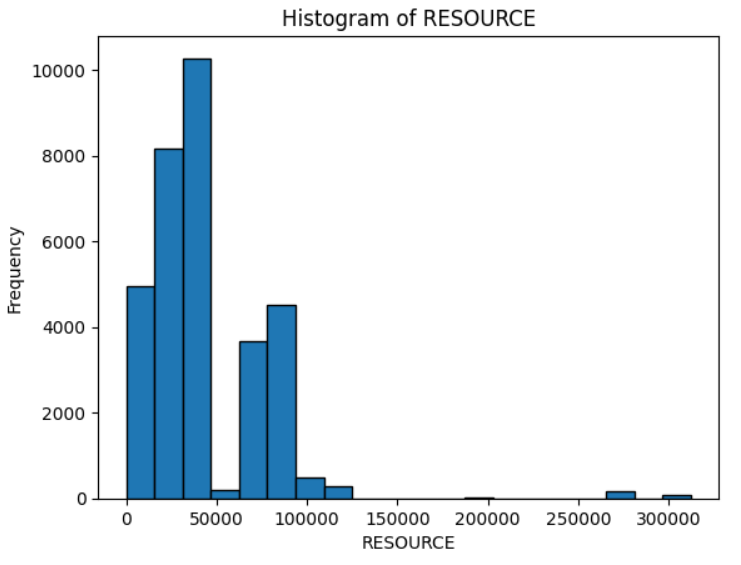
Bộ dữ liệu chứa các thuộc tính sau:

* id: số thứ tự trường dữ liệu.
* RESOURCE: đây là ID cho từng tài nguyên.
* MGR\_ID: đây là ID NHÂN VIÊN của người quản lý của bản ghi ID NHÂN VIÊN hiện tại. Một nhân viên chỉ có thể có một người quản lý tại một thời điểm.
* ROLE\_ROLLUP\_1: Đây là ID danh mục nhóm vai trò của công ty 1.
* ROLE\_ROLLUP\_2: Đây là ID danh mục nhóm vai trò của công ty 2.
* ROLE\_DEPTNAME: Đây là mô tả phòng ban theo vai trò của công ty.
* ROLE\_TITLE: Đây là mô tả tiêu đề doanh nghiệp theo vai trò của công ty.
* ROLE\_FAMILY\_DESC: Đây là mô tả mở rộng về họ vai trò của công ty.
* ROLE\_FAMILY: Đây là mô tả về họ vai trò của công ty.
* ROLE\_CODE: Đây là mã vai trò của công ty. Mã này là duy nhất cho từng vai trò.
* target (nhãn): Giá trị 1 hoặc 0 tương ứng với quyền truy cập được từ chối hay không?

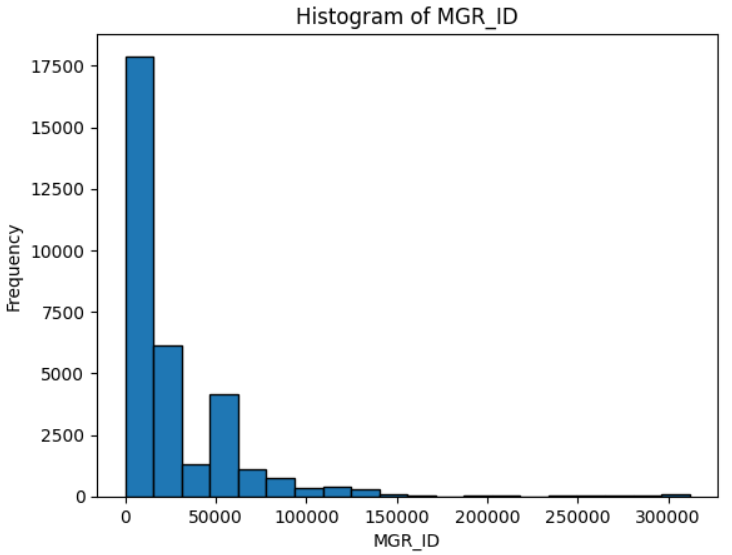
Bộ dữ liệu là một nguồn tài nguyên có giá trị để đào tạo các mô hình học máy để dự đoán sự chấp thuận/từ chối quyền truy cập của nhân viên. Bộ dữ liệu được ghi chép rõ ràng và chứa nhiều tính năng có thể được sử dụng để đưa ra dự đoán chính xác.

## 2. Phân tích dữ liệu và tiền xử lý dữ liệu

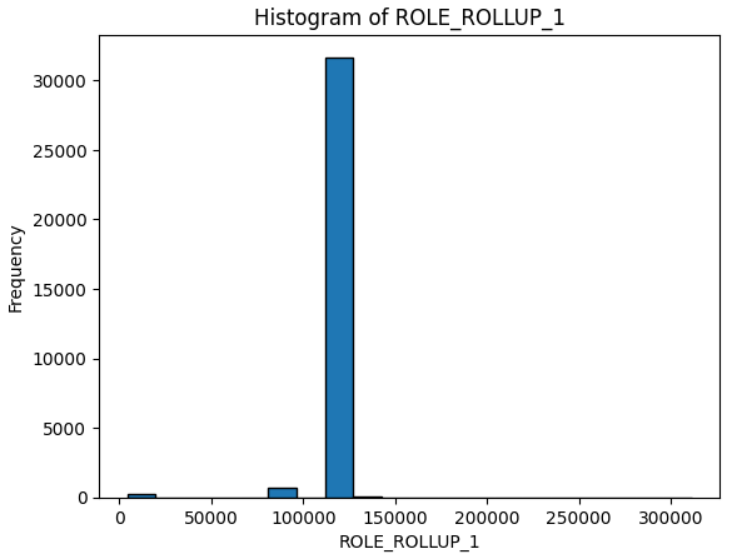
### 2.1. Phân tích dữ liệu

* RESOURCE

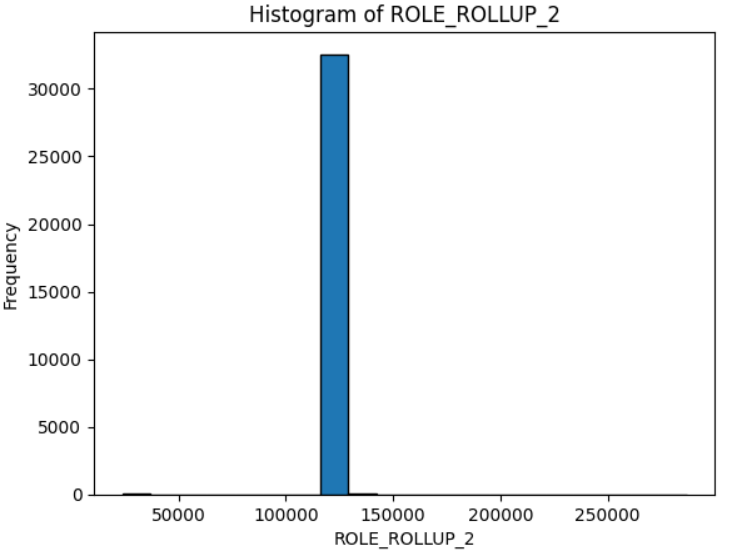
Hình 2: Histogram của thuộc tính RESOURCE

* MGR\_ID

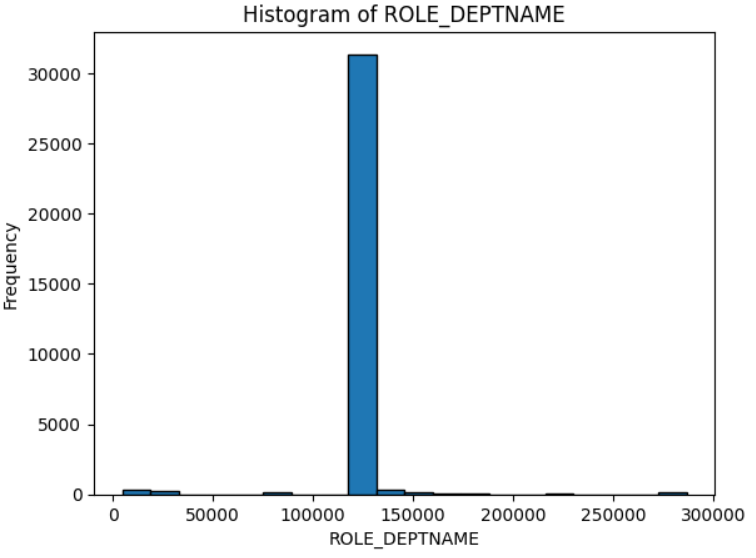
Hình 3: Histogram của thuộc tính MGR\_ID

* ROLE\_ROLLUP\_1

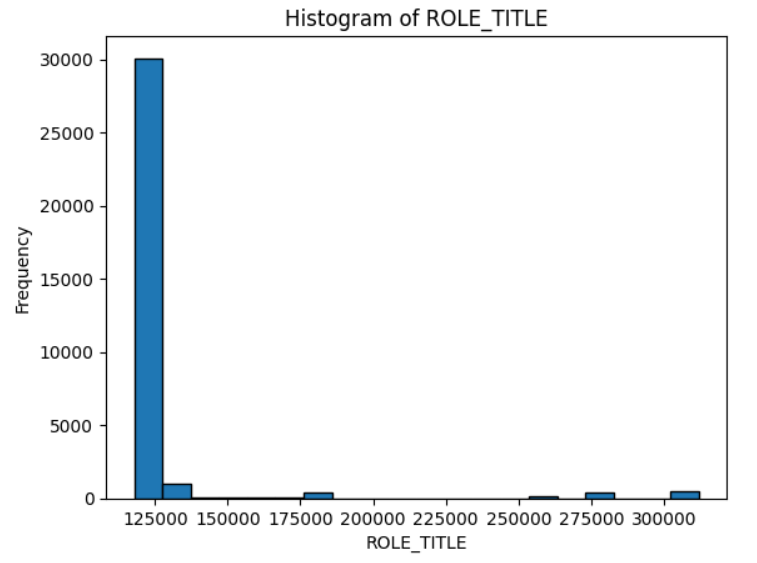
Hình 4: Histogram của thuộc tính ROLE\_ROLLUP\_1

* ROLE\_ROLLUP\_2

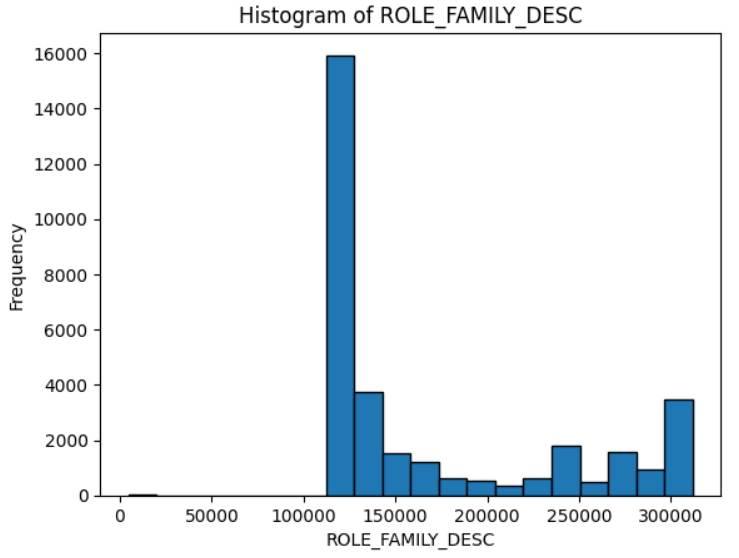
Hình 5: Histogram của thuộc tính ROLE\_ROLLUP\_2

* ROLE\_DEPTNAME

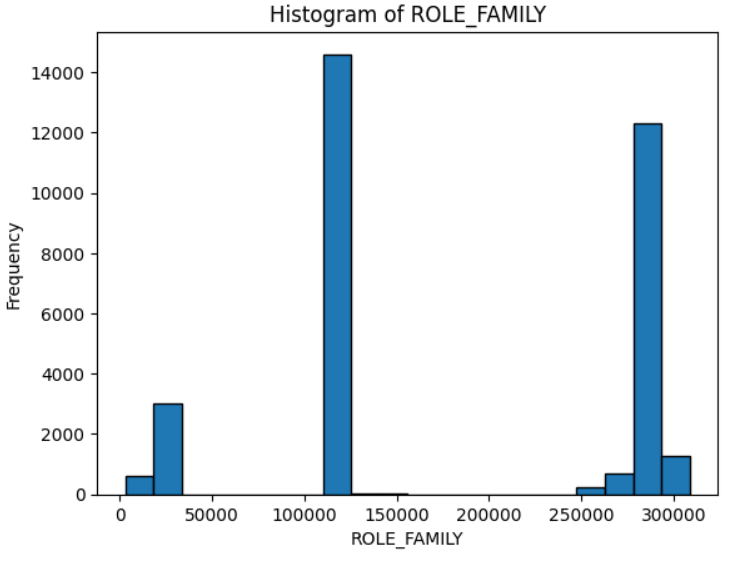
Hình 6: Histogram của thuộc tính ROLE\_DEPTNAME

* ROLE\_TITLE

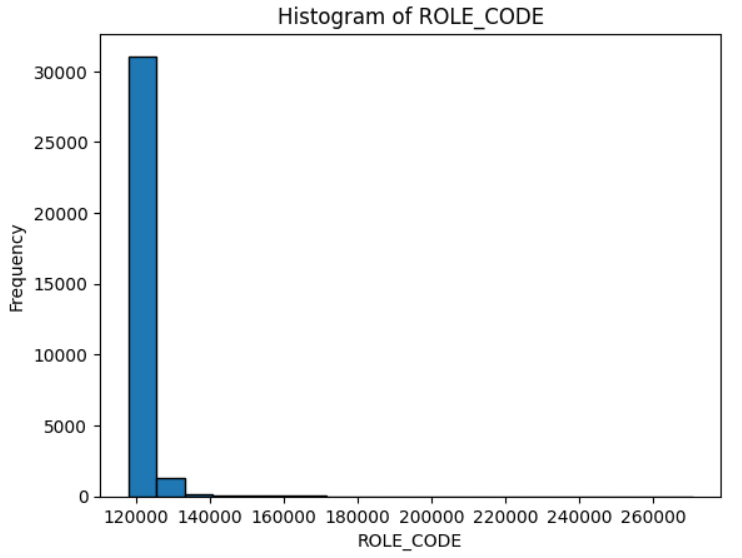
Hình 7: Histogram của thuộc tính ROLE\_TITLE

* ROLE\_FAMILY\_DESC

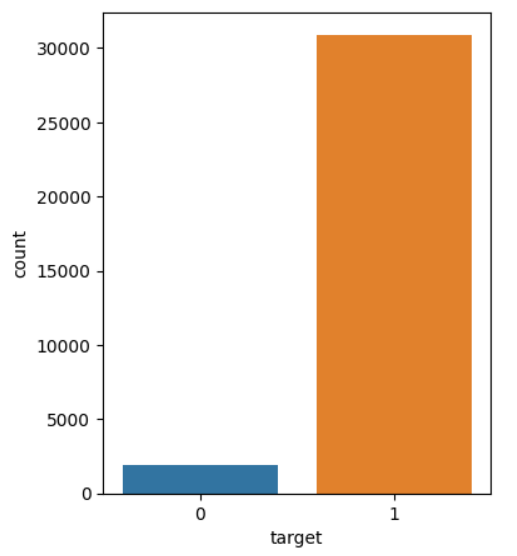
Hình 8: Histogram của thuộc tính ROLE\_FAMILY\_DESC

* ROLE\_FAMILY

Hình 9: Histogram của thuộc tính ROLE\_FAMILY

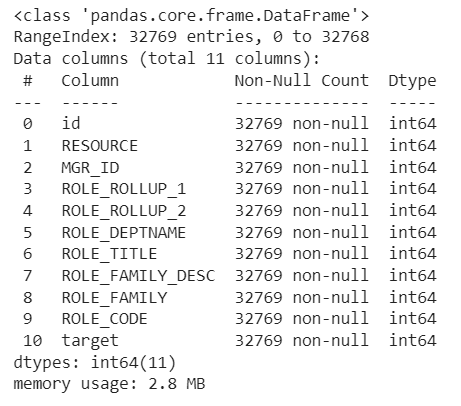
* ROLE\_CODE

Hình 10: Histogram của thuộc tính ROLE\_CODE

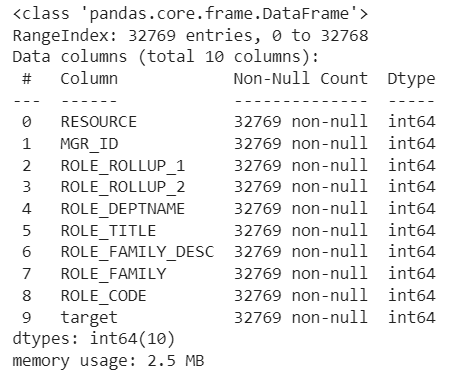
* Cột nhãn (target)

*Nhận xét: có sự mất cấn bằng dữ liệu ở cột nhãn*

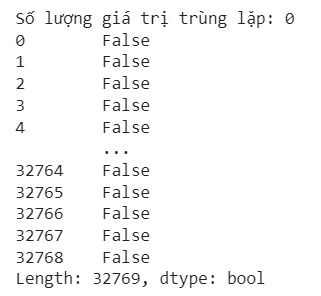
### 2.2. Tiền xử lý dữ liệu

* **Kiểm tra ô trống:** dữ liệu không chứa ô trống.

Hình 11: kết quả tóm tắt thông tin về dataframe

* **Xóa thuộc tính không cần thiết:** xóa cột id, cột id là số thứ tự không có mối tương quan với các thuộc tính khác.

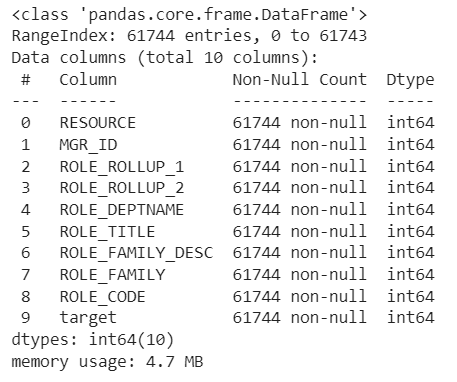
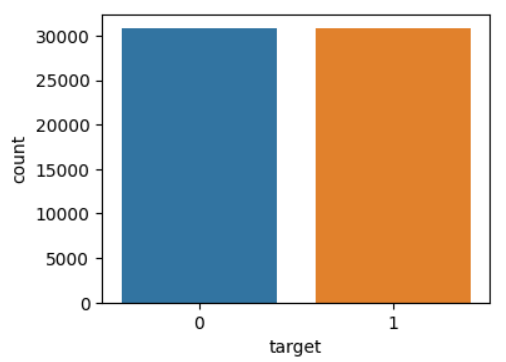
Hình 12: kết quả dataframe sau khi xóa cột id

* **Kiểm tra các mẫu bị trùng lặp:** không có mẫu nào bị trùng lặp.

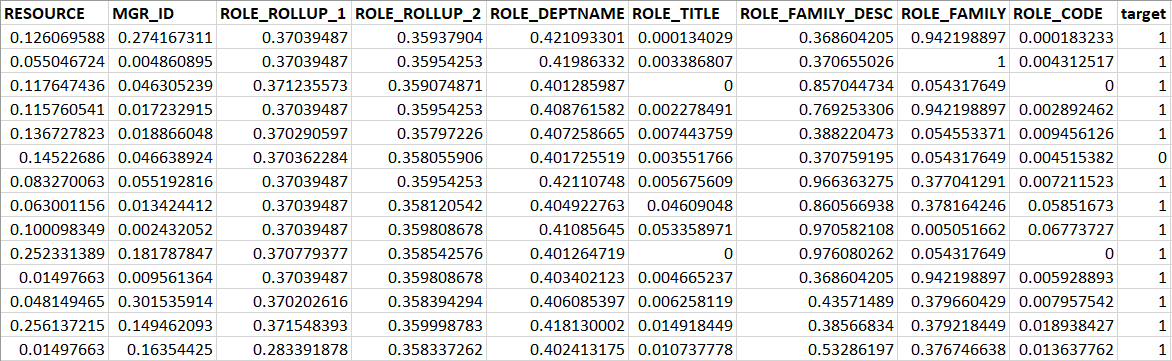
Hình 13: kết quả kiểm tra các mẫu có bị trùng lặp không?

* **Cân bằng dữ liệu:**

Mất cân bằng dữ liệu có thể dẫn đến các vấn đề như hiệu suất kém trên lớp thiểu số hoặc phương sai mô hình cao => Sử dụng phương pháp SMOTE để cân bằng lại dữ liệu.

 SMOTE (Synthetic Minority Over-sampling Technique) là một phương pháp oversampling (tăng cường) dữ liệu được sử dụng để cân bằng mẫu giữa các lớp trong bài toán phân loại, đặc biệt là khi dữ liệu là mất cân bằng (imbalanced). SMOTE tạo ra các mẫu nhân tạo cho lớp thiểu số bằng cách kết hợp thông tin từ các mẫu hiện có của lớp đó. Điều này có thể giúp mô hình học tốt hơn trên lớp thiểu số và cải thiện hiệu suất của mô hình trong tình huống mất cân bằng.

Hình 14: kết quả dataframe sau khi cân bằng lại dữ liệu

* **Chuẩn hóa dữ liệu:** sử dụng phương pháp Min Max Scaling.

Hình 15: dataset sau khi chuẩn hóa dữ liệu

* Hình 15 là kết quả dataset sau giai đoạn tiền xử lý dữ liệu.

## 3. Cấu hình máy tính huấn luyện mô hình

* **Máy tính tiến hành huấn luyện và đánh giá mô hình**
* Công nghệ sử dụng: Google Colab
* GPU thường là GPU Tesla K80
* Bộ nhớ GDDR5: 12 GB
* Tốc độ xung nhịp: 1.4 GHz
* **Máy tính tiến hành triển khai mô hình thành ứng dụng**

Bảng 2: Bảng cấu hình máy tính triển khai mô hình

|  |  |
| --- | --- |
|  | **Phần cứng** |
| **Máy 1** | CPU: Intel Core i7 1165G  RAM: 8GB  Ổ cứng: 512GB SSD |
| **Máy 2** | CPU: Intel Core(TM) i7-1065G7  RAM: 16GB  Ổ cứng: 512GB SSD |

# CHƯƠNG II. HUẤN LUYỆN MÔ HÌNH

## 1. Giải thuật KNN

### 1.1. Giới thiệu

Giải thuật KNN (K-Nearest Neighbors) là một thuật toán học máy có giám sát, được sử dụng để phân loại dữ liệu mới dựa trên thông tin của các dữ liệu đã được phân loại trước đó. Thuật toán này dựa trên giả thuyết rằng các dữ liệu tương tự nhau sẽ nằm gần nhau trong không gian đặc trưng.

### 1.2. Cách hoạt động của giải thuật

Giải thuật KNN hoạt động như sau:

* Đầu tiên, thuật toán sẽ xây dựng một tập huấn luyện gồm các dữ liệu đã được phân loại.
* Khi có một dữ liệu mới cần được phân loại, thuật toán sẽ tính khoảng cách giữa dữ liệu mới đó và tất cả các dữ liệu trong tập huấn luyện.
* Sau đó, thuật toán sẽ tìm ra K dữ liệu gần nhất với dữ liệu mới.
* Cuối cùng, thuật toán sẽ gán nhãn cho dữ liệu mới bằng nhãn của K dữ liệu gần nhất.

### 1.3. Ưu điểm, nhược điểm của giải thuật

* **Ưu điểm:**
* Đơn giản, dễ hiểu, dễ triển khai.
* Không cần thiết phải xây dựng mô hình phức tạp.
* Có thể áp dụng cho các tập dữ liệu có kích thước lớn.
* **Nhược điểm:**
* Độ chính xác của thuật toán phụ thuộc vào tham số k.
* Có thể bị ảnh hưởng bởi các giá trị ngoại lệ.

### 1.4. Kết quả huấn luyện mô hình

* **Kết quả huấn luyện mô hình KNN theo nghi thức Hold-out, datatrain 80% và datatest 20%**

Bảng 3: Bảng kết quả huấn luyện mô hình KNN nghi thức hold-out

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | **Precision** | **Recall** | **F1 score** | **Training time** | **Inference time** |
| KNN | K = 3 | 0.906 | 0.9 | 0.899 | 0.095 | 1.27 |
| K = 5 | 0.893 | 0.882 | 0.882 | 0.092 | 1.277 |
| K = 7 | 0.883 | 0.871 | 0.871 | 0.086 | 1.41 |
| K = 9 | 0.876 | 0.862 | 0.861 | 0.095 | 1.44 |
| K = 11 | 0.864 | 0.849 | 0.848 | 0.091 | 1.41 |

* **Kết quả huấn luyện mô hình KNN theo nghi thức K-fold, K = 15**

Bảng 4: Bảng kết quả huấn luyện mô hình KNN nghi thức 15-fold

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | **Precision** | **Recall** | **F1 score** | **Training time** | **Inference time** |
| KNN | K = 3 | 0.917 | 0.912 | 0.912 | 0.126 | 0.471 |
| K = 5 | 0.903 | 0.895 | 0.894 | 0.121 | 0.534 |
| K = 7 | 0.892 | 0.882 | 0.881 | 0.114 | 0.468 |
| K = 9 | 0.883 | 0.871 | 0.87 | 0.081 | 0.394 |
| K = 11 | 0.876 | 0.862 | 0.861 | 0.15 | 0.596 |

## 2. Giải thuật Naïve Bayes

### 2.1. Giới thiệu

Giải thuật Naive Bayes là một thuật toán phân loại dựa trên xác suất, sử dụng định lý Bayes để tính xác suất của một điểm dữ liệu thuộc một lớp nhất định. Thuật toán này dựa trên giả định rằng các thuộc tính của một điểm dữ liệu là độc lập với nhau.

### 2.2. Cách hoạt động của giải thuật

Giải thuật Naïve Bayes hoạt động như sau:

* Đầu tiên, thuật toán sẽ xây dựng một tập huấn luyện gồm các điểm dữ liệu đã được phân loại.
* Khi có một điểm dữ liệu mới cần được phân loại, thuật toán sẽ tính xác suất của điểm dữ liệu đó thuộc từng lớp.
* Cuối cùng, thuật toán sẽ gán nhãn cho điểm dữ liệu mới bằng lớp có xác suất cao nhất.

### 2.3. Ưu điểm, nhược điểm của giải thuật

* **Ưu điểm:**
* Đơn giản, dễ hiểu, dễ triển khai.
* Có thể áp dụng cho các tập dữ liệu có kích thước lớn.
* Có thể kết hợp với các kỹ thuật khác để cải thiện độ chính xác.
* **Nhược điểm:**
* Giả định độc lập giữa các thuộc tính có thể không chính xác trong một số trường hợp.
* Độ chính xác của thuật toán có thể bị ảnh hưởng bởi các thuộc tính có giá trị hiếm.

### 2.4. Kết quả huấn luyện mô hình

* **Kết quả huấn luyện mô hình Naïve Bayes theo nghi thức Hold-out, datatrain 80% và datatest 20%**

Bảng 5: Bảng kết quả huấn luyện mô hình Naive Bayes nghi thức hold-out

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Precision** | **Recall** | **F1 score** | **Training time** | **Inference time** |
| Naïve Bayes | 0.547 | 0.518 | 0.434 | 0.026 | 0.005 |

* **Kết quả huấn luyện mô hình Naïve Bayes theo nghi thức K-fold, K = 15**

Bảng 6: Bảng kết quả huấn luyện mô hình Naive Bayes nghi thức 15-fold

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Precision** | **Recall** | **F1 score** | **Training time** | **Inference time** |
| Naïve Bayes | 0.551 | 0.519 | 0.432 | 0.012 | 0.001 |

## 3. Giải thuật Decision Tree

### 3.1. Giới thiệu

Giải thuật Decision Tree là một thuật toán học máy có giám sát, được sử dụng để phân loại dữ liệu. Thuật toán này xây dựng một cây quyết định, trong đó mỗi nút đại diện cho một thuộc tính của dữ liệu và mỗi nhánh đại diện cho một giá trị có thể có của thuộc tính đó.

### 3.2. Cách hoạt động của giải thuật

Giải thuật Decision Tree hoạt động như sau:

* Đầu tiên, thuật toán sẽ xây dựng một tập huấn luyện gồm các điểm dữ liệu đã được phân loại.
* Sau đó, thuật toán sẽ bắt đầu từ nút gốc của cây quyết định và sẽ phân chia dữ liệu trong nút đó thành hai hoặc nhiều nhánh.
* Tiếp theo, thuật toán sẽ lặp lại quy trình này cho mỗi nhánh, cho đến khi tất cả dữ liệu trong các nhánh đều được phân loại.

### 3.3. Ưu điểm, nhược điểm của giải thuật

* **Ưu điểm:**
* Dễ hiểu và dễ giải thích.
* Có thể áp dụng cho nhiều loại dữ liệu khác nhau.
* Có thể được áp dụng cho cả bài toán phân lớp và hồi quy.
* **Nhược điểm:**
* Có thể bị quá phức tạp, dẫn đến overfitting.
* Có thể bị ảnh hưởng bởi các dữ liệu ngoại lệ.

### 3.4. Kết quả huấn luyện mô hình

* **Kết quả huấn luyện mô hình Decision Tree theo nghi thức Hold-out, datatrain 80% và datatest 20%**

Bảng 7: Bảng kết quả huấn luyện mô hình Decision Tree nghi thức Hold-out

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | **Precision** | **Recall** | **F1 score** | **Training time** | **Inference time** |
| Decision  Tree | criterion=’gini’, max\_depth=100, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.931 | 0.931 | 0.931 | 0.364 | 0.004 |
| criterion=’gini’, max\_depth=200, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.93 | 0.93 | 0.93 | 0.851 | 0.011 |
| criterion=’gini’, max\_depth=300, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.93 | 0.93 | 0.93 | 0.414 | 0.005 |

* **Kết quả huấn luyện mô hình Decision Tree theo nghi thức K-fold, K = 15**

Bảng 8: Bảng kết quả huấn luyện mô hình Decision Tree nghi thức 15-fold

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | **Precision** | **Recall** | **F1 score** | **Training time** | **Inference time** |
| Decision  Tree | criterion=’gini’, max\_depth=100, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.934 | 0.934 | 0.934 | 0.406 | 0.002 |
| criterion=’gini’, max\_depth=200, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.934 | 0.934 | 0.934 | 0.664 | 0.006 |
| criterion=’gini’, max\_depth=300, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.934 | 0.934 | 0.934 | 0.547 | 0.003 |

## 4. Giải thuật Random Forest

### 4.1. Giới thiệu

Giải thuật Random Forest là một thuật toán học máy có giám sát, được sử dụng để phân loại và hồi quy dữ liệu. Thuật toán này xây dựng một tập hợp các cây quyết định (decision tree), sau đó sử dụng kết quả bỏ phiếu của các cây quyết định này để đưa ra dự đoán.

### 4.2. Cách hoạt động của giải thuật

Giải thuật Random Forest hoạt động như sau:

* Đầu tiên, thuật toán sẽ xây dựng một tập huấn luyện gồm các điểm dữ liệu đã được phân loại.
* Sau đó, thuật toán sẽ xây dựng một số cây quyết định, mỗi cây sẽ được xây dựng trên một tập con ngẫu nhiên của tập huấn luyện.
* Khi có một điểm dữ liệu mới cần được phân loại hoặc dự đoán, thuật toán sẽ sử dụng kết quả bỏ phiếu của các cây quyết định để đưa ra dự đoán.

### 4.3. Ưu điểm, nhược điểm của giải thuật

* **Ưu điểm:**
* Độ chính xác cao.
* Có khả năng chống overfitting.
* Có thể áp dụng cho nhiều loại dữ liệu khác nhau.
* **Nhược điểm:**
* Có thể bị chậm khi triển khai.
* Có thể khó giải thích.

### 4.4. Kết quả huấn luyện mô hình

* **Kết quả huấn luyện mô hình Random Forest theo nghi thức Hold-out, datatrain 80% và datatest 20%**

Bảng 9: Bảng kết quả huấn luyện mô hình Random Forest nghi thức hold-out

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | **Precision** | **Recall** | **F1 score** | **Training time** | **Inference time** |
| Random  Forest | n\_estimators=200,  criterion=’gini’, max\_depth=100, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.962 | 0.962 | 0.962 | 18.312 | 0.461 |
| n\_estimators=300,  criterion=’gini’, max\_depth=100, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.962 | 0.961 | 0.961 | 26.738 | 0.618 |
| n\_estimators=300,  criterion=’gini’, max\_depth=200, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.962 | 0.961 | 0.961 | 26.369 | 0.632 |
| n\_estimators=200,  criterion=’gini’, max\_depth=200, min\_samples\_split=5,  min\_samples\_leaf=10,  random\_state=42 | 0.954 | 0.954 | 0.954 | 17.281 | 0.403 |

* **Kết quả huấn luyện mô hình Random Forest theo nghi thức K-fold, K = 15**

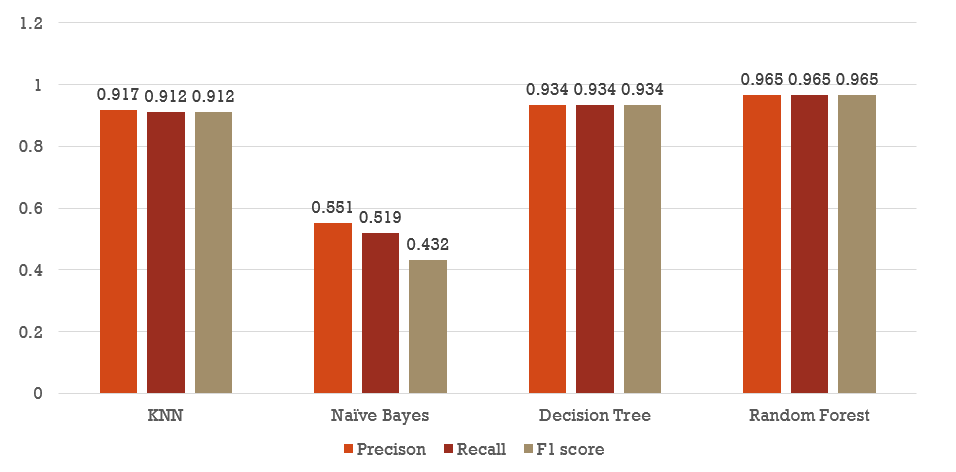
Bảng 10: Bảng kết quả huán luyện mô hình Random Forest nghi thức 15-fold

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | **Precision** | **Recall** | **F1 score** | **Training time** | **Inference time** |
| Random  Forest | n\_estimators=200,  criterion=’gini’, max\_depth=100, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.965 | 0.965 | 0.965 | 20.054 | 0.155 |
| n\_estimators=300,  criterion=’gini’, max\_depth=100, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.965 | 0.965 | 0.965 | 30.413 | 0.253 |
| n\_estimators=300,  criterion=’gini’, max\_depth=200, min\_samples\_split=3,  min\_samples\_leaf=6,  random\_state=42 | 0.965 | 0.965 | 0.965 | 30.409 | 0.246 |
| n\_estimators=200,  criterion=’gini’, max\_depth=200, min\_samples\_split=5,  min\_samples\_leaf=10,  random\_state=42 | 0.958 | 0.957 | 0.957 | 19.699 | 0.156 |

# CHƯƠNG III. ĐÁNH GIÁ VÀ TRIỂN KHAI MÔ HÌNH

## 1. Đánh giá mô hình phân lớp

Hình 16: Kết quả độ chính xác của từng mô hình theo nghi thức đánh giá hold-out

****

Hình 17: Kết quả độ chính xác của từng mô hình theo nghi thức đánh giá 15-fold

Ở cả hai nghi thức đánh giá, mô hình Naïve Bayes đều cho ra kết quả độ chính xác thấp nhất, các mô hình còn lại: KNN, Decision Tree, Random Forest đều cho ra kết quả độ chính xác cao (> 89%). Mô hình Random Forest cho ra độ chính xác cao nhất: 96.2% (nghi thức hold-out) và 96.5% (nghi thức k-fold) => Lựa chọn nghi thức k-fold, mô hình Random Forest để triển khai mô hình thành ứng dụng.

## 2. Nhận xét kết quả thực nghiệm

Hình 18: Ma trận nhầm lẫn mô hình Random Forest

Mô hình ở hình 18 dự đoán ở lớp YES đúng được 1987/2053 mẫu. Mô hình dự đoán lớp NO đúng được 2009/2063 mẫu. Mô hình trên dự đoán đúng được tất cả 3996/4116 mẫu => Độ chính xác của mô hình trên là 97%.

## 3. Triển khai mô hình

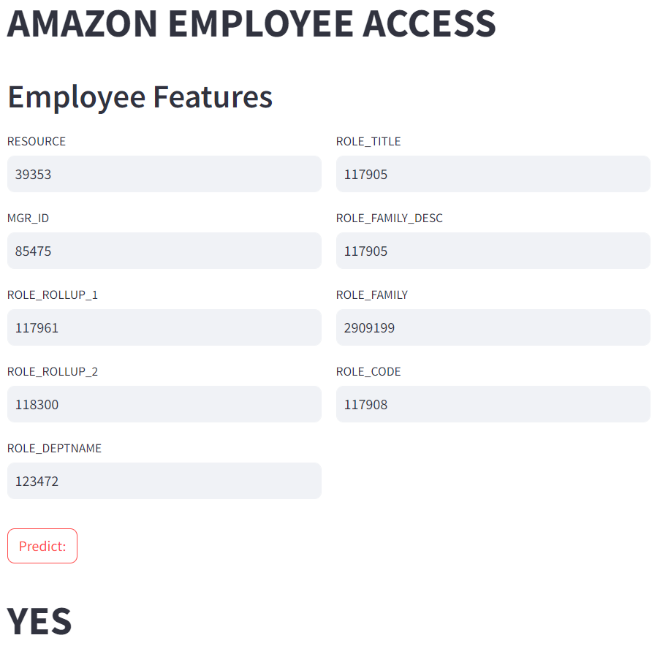
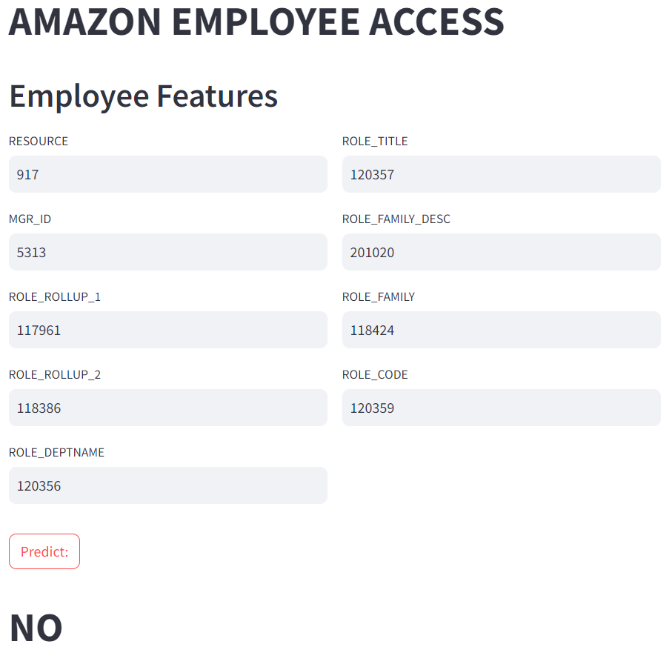
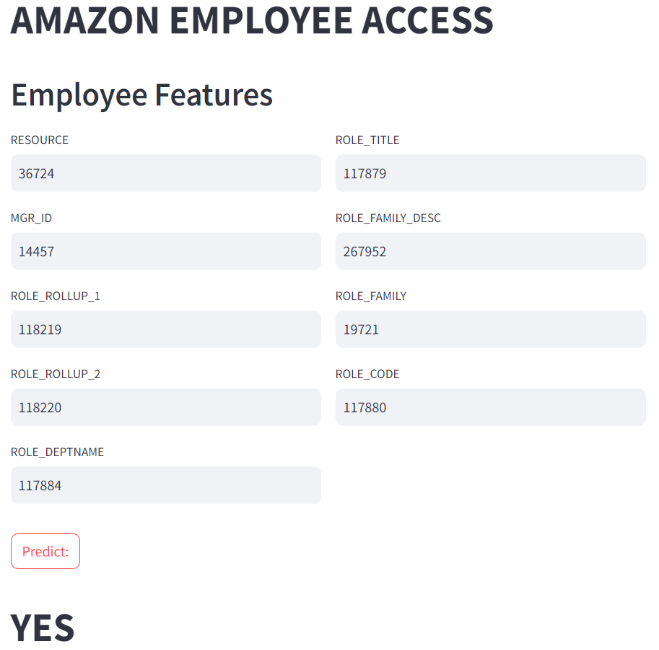
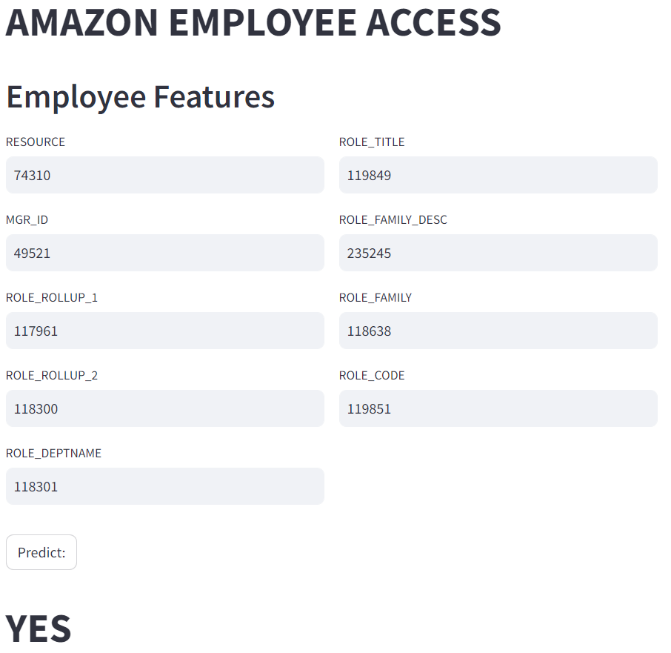
Mô hình máy học được phát triển để giải quyết các vấn đề thực tế và cụ thể trong lĩnh vực nào đó. Bằng cách triển khai mô hình thành ứng dụng, chúng ta có thể áp dụng giải pháp vào môi trường thực tế để cải thiện hoặc tối ưu hóa các quy trình và quyết định. Khi một mô hình máy học được tích hợp vào một ứng dụng, người dùng cuối có thể trực tiếp tận hưởng lợi ích mà mô hình mang lại mà không cần phải hiểu rõ về cách nó hoạt động bên trong. Điều này làm cho công nghệ máy học trở nên dễ sử dụng hơn và có thể áp dụng rộng rãi.

Thư viện hổ trợ triển khai mô hình:

* **Thư viện streamlit:** Streamlit là một thư viện Python mã nguồn mở được sử dụng để tạo các ứng dụng web tương tác một cách nhanh chóng và dễ dàng. Thư viện này cung cấp một bộ công cụ GUI mạnh mẽ cho phép các nhà khoa học dữ liệu, nhà phân tích dữ liệu và nhà phát triển web tạo các ứng dụng web đẹp mắt và hiệu quả chỉ với một vài dòng mã.
* **Thư viện joblib:** Joblib là một thư viện Python mã nguồn mở được sử dụng để lưu trữ và phục hồi các đối tượng Python. Thư viện này dựa trên NumPy và SciPy, và nó cung cấp các phương pháp hiệu quả để lưu trữ các đối tượng Python dưới dạng các tệp nhị phân. Có thể sử dụng joblib để lưu mô hình máy học.
* Bảng test case kiểm thử ứng dụng:

Bảng 11: Bảng test case kiểm thử ứng dụng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case** | **Dữ liệu kiểm thử** | **Kết quả mong đợi** | **Kết quả thực tế** | **Pass/ Fail** |
| TC01 | resource=39353, mgr\_id=85475, role\_rollup\_1=117961, role\_rollup\_2=118300, role\_deptname=123472, role\_title=117905, role\_family\_desc=117905, role\_family=2909199, role\_code=117908 | YES | YES | Pass |
| TC02 | resource=917, mgr\_id=5313, role\_rollup\_1=117961, role\_rollup\_2=118386, role\_deptname=120356, role\_title=120357, role\_family\_desc=201020, role\_family=118424, role\_code=120359 | NO | NO | Pass |
| TC03 | resource=36724, mgr\_id=14457, role\_rollup\_1=118219, role\_rollup\_2=118220, role\_deptname=117884, role\_title=117879, role\_family\_desc=267952, role\_family=19721, role\_code=117880 | YES | YES | Pass |
| TC04 | resource=74310, mgr\_id=49521, role\_rollup\_1=117961, role\_rollup\_2=118300, role\_deptname=118301, role\_title=119849, role\_family\_desc=235245, role\_family=118638, role\_code=119851 | NO | YES | Fail |



Hình 19: Kết quả kiểm thử mô hình đã được triển khai

# PHẦN KẾT LUẬN

## **1. Kết quả đạt đượ**c

* Xây dựng mô hình máy học với độ chính xác 96%.
* Triển khai thành công mô hình thành ứng dụng với sự hỗ trợ của thư viện Streamlit.

## 2. Hướng phát triển

* Triển khai mô hình chạy trên nền tảng website/ mobile app.
* Tích hợp mô hình máy học vào hệ thống nhúng.

# TÀI LIỆU THAM KHẢO

[1]. Andreas C. Muller & Sarah Guido, Introduction to Machine Learning with Python, O'Reilly Media, 2016.

[2]. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[3]. Andrew Ng, Machine Learning Yearning, <https://github.com/ajaymache/machine-learning-yearning>