

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG



BÁO CÁO ĐỒ ÁN

Môn học: Đồ án chuyên ngành

**Đề tài: Phát triển hệ thống MLOps cho ứng dụng dự đoán
kết quả học tập của học sinh**

EduPredictOps

Giáo viên hướng dẫn: TS. Nguyễn Tấn Hoàng Phước

Thành viên:

Phạm Minh Thuận - 21522651

NĂM HỌC : 2024-2025

Mục lục

CHƯƠNG I: GIỚI THIỆU TỔNG QUAN VỀ ĐỀ TÀI.....	4
1. Ý tưởng:.....	4
2. Giới thiệu sơ lược đề tài:	4
3. Mục đích:.....	4
4. Các công cụ hỗ trợ:.....	5
5. Phương pháp nghiên cứu.....	5
CHƯƠNG II: CƠ SỞ LÝ THUYẾT.....	5
1. Python Flask-app	5
2. Model Machine Learning	6
3. Flask Blueprint API:.....	10
4. Đánh giá hiệu suất mô hình (Model Evaluation):	11
5. Triển khai ứng dụng:	11
CHƯƠNG III: PHÂN TÍCH THIẾT KẾ HỆ THỐNG	13
1. Yêu cầu:.....	13
2. Đặc tả yêu cầu:	13
3. Đặc tả USE CASE:.....	15
4. Sơ đồ phân rã:	15
CHƯƠNG IV: KẾT QUẢ THỰC HIỆN	17
1. Trang web dành cho người dùng:.....	17
2. Trang web dành cho admin:	20
CHƯƠNG V: KẾT LUẬN.....	25
1. Kết quả đạt được.....	25
2. Hạn chế:.....	25
3. Hướng phát triển dự án:.....	26
Tài liệu tham khảo.....	26

Danh sách ảnh:

Hình 1: Sơ đồ Perceptron đa lớp	7
Hình 2: Sơ đồ kiến trúc mạng nơ hồi qui LSTM.....	8
Hình 3: Sơ đồ hệ thống tổng quát.	15
Hình 4: Giao diện nhập liệu trực tiếp.	17
Hình 5: Giao diện nhập liệu bằng file excel	18
Hình 6: Giao diện hiển thị kết quả dự đoán trực tiếp.	19
Hình 7: Giao diện hiển thị kết quả dự đoán bằng excel.	19
Hình 8: Giao diện mô tả dữ liệu.	20
Hình 9: Giao diện đăng nhập.....	21
Hình 10: Giao diện đổi mật khẩu.	21
Hình 11: Giao diện hiển thị hiệu suất.....	22
Hình 12: Giao diện nhập liệu cho quá trình train model.	23
Hình 13: Giao diện so sánh độ tin cậy của model.	24
Hình 14: Giao diện mô tả kết quả sau khi train.....	25

CHƯƠNG I: GIỚI THIỆU TỔNG QUAN VỀ ĐỀ TÀI

1. Ý tưởng:

- Đề tài hướng đến việc tích hợp và mở rộng dữ liệu cho các mô hình Machine Learning trong ứng dụng web, đồng thời cung cấp khả năng quản lý hiệu suất và đánh giá chất lượng hoạt động của mô hình. Điều này giúp người quản trị đưa ra các giải pháp cải thiện hiệu suất trong quá trình triển khai, đồng thời hỗ trợ việc ra quyết định dựa trên dữ liệu cập nhật.

2. Giới thiệu sơ lược đề tài:

- **EduPredictOps** là một ứng dụng web tích hợp 3 mô hình Machine Learning nhằm dự đoán kết quả học tập của học sinh. Ứng dụng không chỉ tập trung vào khả năng dự đoán mà còn cung cấp các tính năng quản lý hiệu suất, giúp người quản trị theo dõi và đánh giá chất lượng của các mô hình.
- Ngoài ra, ứng dụng hỗ trợ việc nâng cấp và mở rộng các mô hình thông qua việc bổ sung tập dữ liệu mới. Người quản trị có thể đánh giá mức độ ảnh hưởng của tập dữ liệu được cập nhật đến hiệu suất dự đoán, từ đó quyết định chấp nhận hoặc từ chối dữ liệu mới. Đây là một bước tiến quan trọng trong việc cải thiện khả năng dự đoán và tối ưu hóa mô hình Machine Learning trong môi trường thực tiễn.

3. Mục đích:

- Nghiên cứu công nghệ và lựa chọn mô hình tối ưu: Tìm hiểu, đánh giá, và áp dụng các mô hình Machine Learning phù hợp nhất để dự đoán chính xác kết quả học tập của học sinh.
- Phát triển ứng dụng web EduPredictOps: Tạo ra một ứng dụng thực tiễn, hỗ trợ học sinh và giảng viên trong việc đánh giá, đưa ra chiến lược học tập và giảng dạy phù hợp với từng đối tượng học sinh.
- EduPredictOps tập trung vào:
 - **Dự đoán kết quả học tập:** Sử dụng dữ liệu đầu vào như điểm số trong quá khứ và các yếu tố hoàn cảnh gia đình (mồ côi, gia đình khó khăn).

Cải thiện mô hình: Cung cấp khả năng nâng cấp và tối ưu mô hình thông qua cập nhật dữ liệu (*chỉ dành cho người quản trị).

Đánh giá hiệu suất: Hiển thị thông số chính xác của mô hình để hỗ trợ người quản trị đưa ra các quyết định liên quan đến việc cập nhật dữ liệu.

4. Các công cụ hỗ trợ:

- IDE: Visual Studio Code,
- Framework: Python, Flask, HTML, JavaScript, SQLAlchemy
- API: Flask Blueprint API
- Other: Docker Desktop, Azure Cloud.

5. Phương pháp nghiên cứu

- Tìm hiểu mô hình Machine Learning: Nghiên cứu cách hoạt động và lựa chọn các mô hình phù hợp nhất.
- Tham khảo tài liệu: Tiếp cận các tài liệu về huấn luyện mô hình, xử lý dữ liệu, và đánh giá hiệu suất.
- Xây dựng hệ thống phân quyền: Nghiên cứu luồng hoạt động và thiết kế hệ thống phân quyền giữa người dùng và người quản trị để áp dụng vào dự án.

EduPredictOps hướng đến việc tận dụng sức mạnh của Machine Learning và MLOps để mang lại giá trị thực tiễn cao, không chỉ hỗ trợ học sinh và giảng viên mà còn hỗ trợ cho người quản trị dễ dàng vận hành và nâng cấp hệ thống.

CHƯƠNG II: CƠ SỞ LÝ THUYẾT

1. Python Flask-app

Flask là một web framework nhẹ (microframework) được viết bằng ngôn ngữ lập trình Python. Nó cung cấp các công cụ và thư viện cần thiết để phát triển ứng dụng web một cách dễ dàng và nhanh chóng. Flask được thiết kế với kiến trúc đơn giản, không bao gồm các thành phần nặng như ORM (Object Relational Manager) hay tính năng quản lý biểu đồ quan hệ.

Flask được phát triển bởi Armin Ronacher, dựa trên các công cụ như Werkzeug WSGI (một thư viện hỗ trợ giao tiếp giữa web server và ứng dụng) và template engine Jinja2. Các đặc điểm nổi bật của Flask bao gồm:

- Dễ dàng sử dụng: Phù hợp với người mới bắt đầu và các dự án nhỏ.
- Mở rộng linh hoạt: Đa dạng thư viện hoặc công cụ dễ dàng tích hợp theo các yêu cầu của dự án.
- Blueprint API: Cho phép phân chia mã nguồn thành các module nhỏ hơn, hỗ trợ phát triển ứng dụng lớn với nhiều phần độc lập.

2. Model Machine Learning

Trong dự án này, có 3 mô hình Machine Learning sau được áp dụng:

2.1. Linear Regression (Hồi quy tuyến tính)(1):

Linear Regression là một thuật toán học máy có giám sát, được sử dụng để tìm mối quan hệ tuyến tính giữa biến phụ thuộc (đầu ra) và một hoặc nhiều biến độc lập (đầu vào). Đây là phương pháp cơ bản để dự đoán các giá trị liên tục.

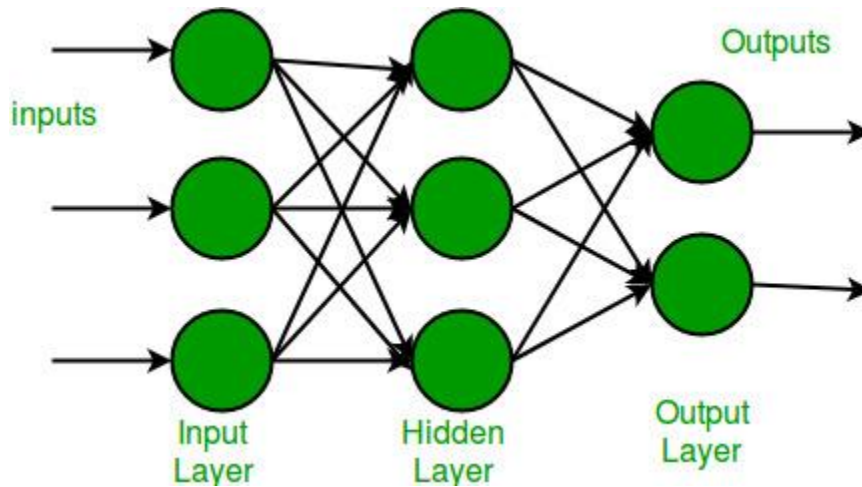
Phương trình hồi quy tuyến tính:

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n$$

- y : Biến phụ thuộc (đầu ra).
- X_1, X_2, \dots, X_n : Các biến độc lập (đầu vào).
- β_0 : Hệ số chặn (intercept).
- $\beta_0, \beta_2, \dots, \beta_n$: Hệ số góc tương ứng với từng biến độc lập.

Ứng dụng: Linear Regression được sử dụng để tìm mối quan hệ đơn giản giữa các biến và dự đoán giá trị đầu ra cho các bài toán không quá phức tạp.

2.2. Multilayer Perceptron (MLP)(2):



Hình 1: Sơ đồ Perceptron đa lớp

MLP là một loại mạng neural nhân tạo (Artificial Neural Network - ANN) bao gồm nhiều lớp fully connected. Mỗi lớp trong MLP áp dụng hàm kích hoạt để học các mối quan hệ phi tuyến giữa đầu vào và đầu ra.

Công thức mô hình MLP:

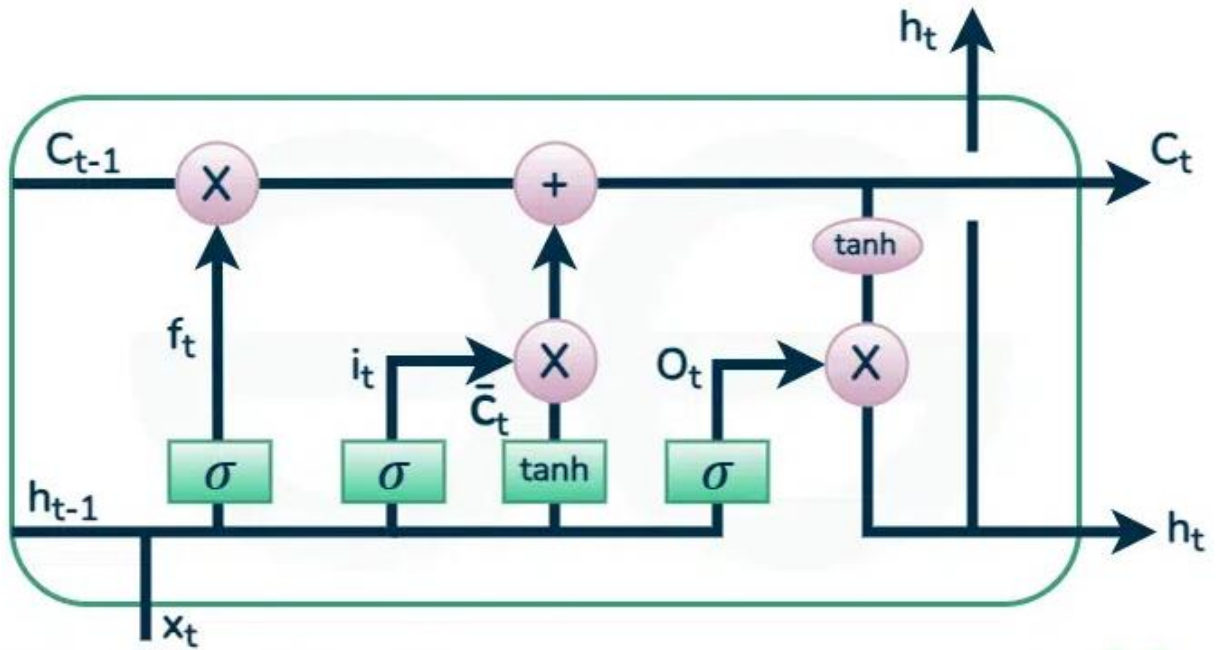
$$a^{(l)} = f(W^{(l)}a^{(l-1)} + b^{(l)})$$

- $a^{(l)}$: Đầu ra của lớp l.
- $W^{(l)}$: Trọng số của lớp l.
- $b^{(l)}$: Hệ số bù của lớp l.
- f : Hàm kích hoạt. Ở đây dùng ReLU ($f(z) = \max(0, z)$) ở 3 lớp đầu và Linear ở lớp thứ 4 $f(z) = z$.

$$\Rightarrow a^{(4)} = W^{(4)}(\text{ReLU}(W^{(3)} \cdot \text{ReLU}(W^{(2)} \cdot \text{ReLU}(W^{(1)}x + b^{(1)}) + b^{(2)}) + b^{(3)}) + b^{(4)})$$

Ứng dụng: MLP có thể được đào tạo bằng tính toán song song, có thể mô hình hóa các mối quan hệ phức tạp, phi tuyến tính trong dữ liệu, có thể được áp dụng cho nhiều vấn đề khác nhau, cả phân loại và hồi quy.

2.3. Long Short-Term Memory (LSTM)(3):



Hình 2: Sơ đồ kiến trúc mạng nơ hồi qui LSTM

LSTM là một biến thể của mạng neural hồi tiếp (Recurrent Neural Network - RNN), được thiết kế để giải quyết vấn đề dự đoán chuỗi (sequence prediction) và nắm bắt các phụ thuộc dài hạn trong dữ liệu.

Các thành phần chính của LSTM:

- **Forget Gate:** Xác định thông tin từ trạng thái trước đó quyết định loại bỏ một số ô nhớ không còn hữu ích.

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$$

- **Input Gate:** Quyết định thông tin nào sẽ được thêm vào ô nhớ, tính toán cho giá trị được cập nhật vào ô nhớ tiếp theo.

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c)$$

- **Cell State Update:** Cập nhật trạng thái ô nhớ, thực hiện quên một phần thông tin cũ qua Forget Gate và thông tin mới từ Input Gate:

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t$$

- **Output Gate:** Quyết định thông tin nào sẽ được xuất ra.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$$

- f_t, i_t, o_t : Giá trị của các cổng Forget, Input, và Output tương ứng tại thời điểm t .
- \tilde{C}_t : giá trị tìm năng cập nhật vào C_t : Trạng thái ô nhớ tại thời điểm t .
- h_t : Trạng thái ẩn (hidden state) tại thời điểm t .
- σ : Hàm kích hoạt sigmoid.
- \tanh : hàm kích hoạt cho \tilde{C}_t .
- W_f, W_i, W_c, W_o : Ma trận trọng số của cổng quên, cổng vào, cổng Cell, cổng ra, ở đây dùng dùng 3 lớp LSTM cho 3 lớp đầu và 1 lớp Dense cho lớp cuối cùng.
- b_f, b_i, b_c, b_o : Hệ số bù của cổng quên, cổng vào, cổng Cell, cổng ra.

$$\Rightarrow h_t^{(1)} = \text{LSTM}_1(x_t)$$

$$\Rightarrow h_t^{(2)} = \text{LSTM}_2(h_t^{(1)})$$

$$\Rightarrow h_t^{(3)} = \text{LSTM}_3(h_t^{(2)})$$

$$\Rightarrow y = W(f_{c1}h_t^{(3)} + b_{fc1})$$

Ứng dụng: LSTM đã được sử dụng cho các tác vụ dự báo chuỗi thời gian như dự đoán giá cổ phiếu, thời tiết và mức tiêu thụ năng lượng,...(ở đây là dự đoán kết quả học tập trong tương lai). LSTM có thể tìm hiểu các mẫu trong dữ liệu chuỗi thời gian và sử dụng chúng để đưa ra dự đoán trong tương lai(ở đây là bộ dữ liệu trong 3 năm liên tục).

2.4. Các thông số trong quá trình train mode

Đối với modal Linear RegressionL hệ số (parameters) trong LR được ước lượng bằng cách giải bài toán tối ưu. Quá trình train model này không cần chia nhỏ dữ liệu hay lặp lại nhiều lần

Đối với 2 mô hình Multilayer Perceptron và Long Short-Term Memory các thông số cần quan tâm là epochs và path_size. Đây là 2 tham số quan trọng trong huấn luyện mô hình.

- **epochs:** Số lần huấn luyện bộ dữ liệu, 1 epochs là 1 lần mô hình học tập từ bộ dữ liệu. Giá trị epochs càng lớn model càng có điều kiện học tập và tối ưu hơn. Nhưng nếu nó quá lớn có thể gây ra tình trạng overfitting gây giảm hiệu suất đối với bộ dữ liệu mới vì mô hình bắt đầu "học thuộc lòng" bộ dữ liệu thay vì tổng quát hóa.
- **bath_size:** Kích thước mẫu dữ liệu được đưa vào model trong một lần cập nhật giá trị. Trong một epochs bộ dữ liệu sẽ được chia nhỏ theo kích thước bath_size để đưa vào model. Giá trị path_size càng lớn các chỉ số mất mát (weights, biases,..) càng ổn định, model càng dễ hội tụ đối lại yêu cầu lớn về bộ nhớ và GPU. Model LSTM cần chỉ số bath_size và epochs lớn hơn so với MLP vì cần học tập từ bộ dữ liệu tuần tự theo chuỗi thời gian.

2.5. Kết luận:

Ba mô hình trên được lựa chọn để xử lý các bài toán trong dự án, bao gồm: hồi quy tuyến tính để tìm mối quan hệ đơn giản giữa đầu vào và đầu ra, MLP cho các bài toán phức tạp hơn với dữ liệu phi tuyến, và LSTM cho các bài toán liên quan đến chuỗi thời gian.

3. Flask Blueprint API:

Blueprint là một tính năng mạnh mẽ của Flask, cho phép tổ chức ứng dụng thành các module độc lập, hỗ trợ việc phát triển và bảo trì các ứng dụng lớn. Với Blueprint, các phần của ứng dụng (ví dụ: routes, templates) có thể được đóng gói riêng lẻ và tái sử dụng trong nhiều dự án khác nhau.

Lợi ích:

- Tăng khả năng mở rộng.
- Cải thiện khả năng bảo trì mã nguồn.

- Dễ dàng quản lý các phần khác nhau trong ứng dụng.

4. Đánh giá hiệu suất mô hình (Model Evaluation):

Đánh giá hiệu suất của mô hình là một bước quan trọng trong việc đảm bảo rằng mô hình học máy đáp ứng các yêu cầu thực tế. Một số tiêu chí và phương pháp phổ biến được sử dụng bao gồm:

- **Mean Absolute Error (MAE):** Trung bình khoảng cách tuyệt đối giữa giá trị thực tế và giá trị dự đoán.

$$MAE = \frac{1}{n} \sum_{i=0}^n |y - y'|$$

- **Mean Absolute Percentage Error (MAPE):** Phần trăm sai số trung bình giữa giá trị thực tế và giá trị dự đoán.

$$MAPE = \frac{1}{n} \sum_{i=0}^n \left| \frac{y - y'}{y} \right| * 100$$

- **Mean Squared Error (MSE):** Tính trung bình của bình phương sai số.

$$MSE = \frac{1}{n} \sum_{i=0}^n (y - y')^2$$

- **Root Mean Squared Error (RMSE):** Căn bậc hai của MSE, thường dễ hiểu hơn do cùng đơn vị với đầu ra.

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n} \sum_{i=0}^n (y - y')^2}$$

- **Accuracy:** Tỷ lệ dự đoán đúng trên tổng số điểm dữ liệu.

$$Accuracy = \frac{\text{Số dự đoán đúng}}{\text{Tổng số dự đoán}} * 100$$

5. Triển khai ứng dụng:

5.1. Docker:

Docker là một công cụ dùng để đóng gói, triển khai và chạy các ứng dụng trong các container. Một số lợi ích khi sử dụng Docker trong phát triển và triển khai mô hình học máy:

- Môi trường nhất quán: Docker giúp đảm bảo rằng mô hình của bạn chạy chính xác trên mọi môi trường (local, staging, production).

- Đóng gói toàn bộ dự án: Bao gồm mô hình, mã nguồn, thư viện, và phụ thuộc.
- Dễ dàng chia sẻ và triển khai: Docker images có thể dễ dàng được chia sẻ qua Docker Hub hoặc các registry riêng.

5.1. Azure/ AWS:

Azure là một nền tảng điện toán đám mây do Microsoft phát triển, cung cấp các dịch vụ như lưu trữ, phân tích dữ liệu, trí tuệ nhân tạo, và triển khai ứng dụng. Với khả năng mở rộng cao, Azure hỗ trợ doanh nghiệp xây dựng, triển khai và quản lý ứng dụng trên quy mô toàn cầu thông qua cơ sở hạ tầng đám mây mạnh mẽ. Azure cung cấp các dịch vụ mạnh mẽ để triển khai và quản lý mô hình học máy

- Azure Machine Learning: Dịch vụ toàn diện để huấn luyện, triển khai và giám sát mô hình.
- Azure Kubernetes Service (AKS): Triển khai các container Docker ở quy mô lớn với tính năng cân bằng tải tự động.
- Azure Blob Storage: Lưu trữ dữ liệu lớn để phục vụ cho quá trình huấn luyện và đánh giá mô hình.

Docker giúp đóng gói ứng dụng WebApp thành một container để đảm bảo môi trường chạy nhất quán trên mọi nền tảng. Sau khi tạo Docker image, nó được push lên Azure Container Registry (ACR) để quản lý và lưu trữ. Từ ACR, ứng dụng có thể dễ dàng triển khai lên Azure App Service, đảm bảo tính linh hoạt, hiệu suất cao và khả năng mở rộng khi phục vụ người dùng. Việc sử dụng Docker và Azure giúp tăng tốc triển khai, giảm thiểu lỗi và tối ưu hoá quy trình phát triển ứng dụng.

CHƯƠNG III: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

1. Yêu cầu:

1.1. Yêu cầu ứng dụng phía người dùng:

- Dự đoán kết quả học tập trong năm lớp 12:
 - Dự đoán thông qua thông tin người dùng nhập.
 - Dự đoán thông qua file excel do người dùng upload.
- Dự đoán kết quả thi trung học phổ thông:
 - Dự đoán thông qua thông tin người dùng nhập.
 - Dự đoán thông qua file excel do người dùng upload.
- Cập nhật thông tin về số lần submit và thông số performance của ứng dụng web đối với mỗi lần tính toán.

1.2. Yêu cầu ứng dụng phía người quản trị:

- Đăng nhập xác thực tài khoản để thực hiện các routes của admin.
- Hiển thị các thông số về hiệu suất của ứng dụng web.
- Train thêm dữ liệu cho các bài toán dự đoán điểm.
- Hiển thị kết quả về độ chính xác của model sau khi train.
- Chấp nhận models mới đã train hoặc từ chối models mới (sử dụng lại models cũ).
- Cập nhật lại bộ dữ liệu và thực hiện lại trực quan hóa dữ liệu.

2. Đặc tả yêu cầu:

2.1. Thực hiện dự đoán:

- Người dùng có thể dự đoán kết quả học tập của mình thông qua dữ liệu học tập được nhập từ trang web.
- Giáo viên có thể dự đoán kết quả của một nhóm học sinh thông qua việc upload file excel lưu trữ dữ liệu của nhóm học sinh, đồng thời có thể tải kết quả dự đoán của 3 model thành file excel để dễ dàng cho việc đánh giá cũng như có phương án hỗ trợ và đưa ra phương án giảng dạy phù hợp

2.2. Cập nhật thông tin về số lần submit và performance:

- Thông tin về số lần submit trong ngày hôm nay và tổng số lần submit.
- Thông tin về latency, mức độ sử dụng của CPU và Ram, số lần predict được trên giây, số lần dự đoán trên 1 request.

2.3. Đăng nhập xác thực quyền admin để có thể truy cập đến các routes admin:

Admin nhập thông tin đăng nhập (đã được đăng ký từ trước - chỉ có 1 admin) để truy cập vào hệ thống.

Chỉ có admin mới có thể thực hiện thay đổi thông tin tài khoản.

2.3. Hiển thị các thông số về hiệu suất của ứng dụng web:

Mỗi lần reload sẽ thực hiện truy vấn đến database để tính toán hiệu suất trung bình của ứng dụng web nhằm đưa ra thông số chính xác nhất cho admin có thể đánh giá mức độ tối ưu trong sử dụng tài nguyên từ đó đưa ra giải pháp mở rộng nếu thiếu tài nguyên hoặc giải thiểu nếu dư thừa tài nguyên.

2.4. Train thêm dữ liệu cho các bài toán dự đoán điểm:

Dữ liệu sau khi được admin xử lý phù hợp với models sẽ được train thêm cho model, kết quả train sẽ tạo ra 2 phiên bản models (model ban đầu và model sau khi train).

Giá trị của các thông số áp dụng trong quá trình train(giá trị epochs, và bath_size) sẽ được tính toán phù hợp dựa trên kích thước của bộ dữ liệu.

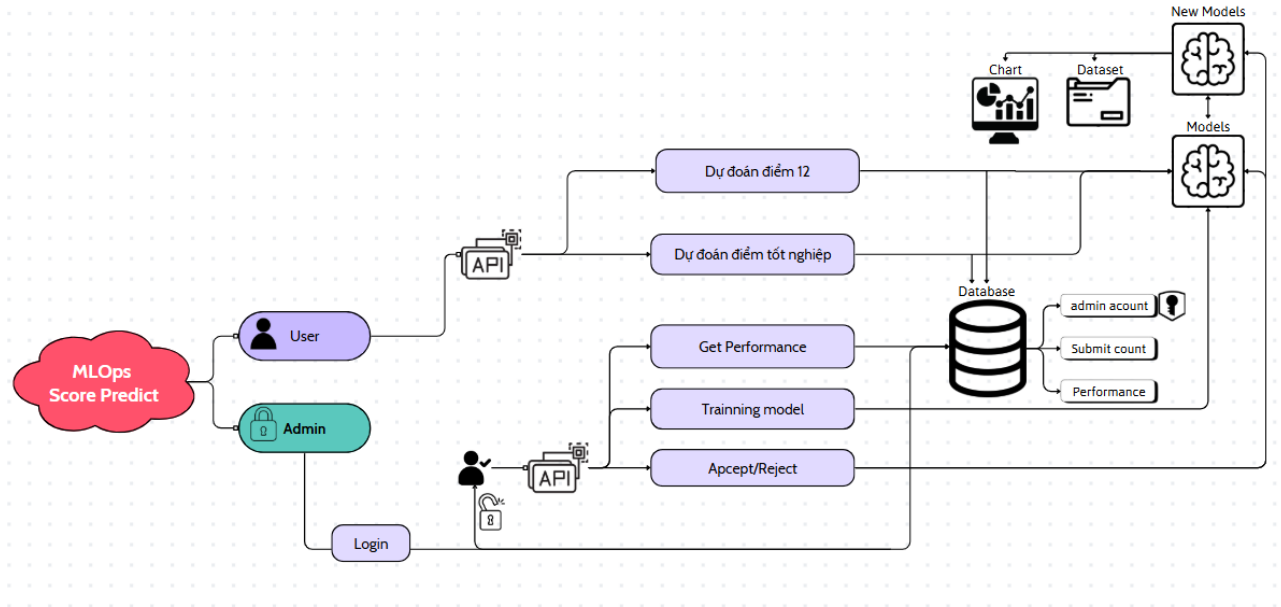
2.5. Hiển thị kết quả về độ chính xác của model sau khi train:

Model sau khi train xong sẽ được tính toán các thông số về độ chính xác: 'MAE', 'MAPE', 'MSE', 'RMSE', 'Accuracy'. Admin có thể đánh giá mức độ tin cậy của model từ đó đưa ra quyết định chấp nhận models mới hay dùng lại models cũ.

2.6. Cập nhật lại bộ dữ liệu và thực hiện lại trực quan hóa dữ liệu:

Trường hợp admin chấp nhận sử dụng model mới, bộ dữ liệu được training cho model sẽ được cập nhật lại. Đồng thời, các tính toán và biểu đồ thể hiện trực quan hóa dữ liệu sẽ được thực hiện lại.

3. Đặc tả USE CASE:



Hình 3: Sơ đồ hệ thống tổng quát.

4. Sơ đồ phân rã:

Nhóm đối tượng	Tính năng chính	Chi tiết
USER	Dự đoán từ bài toán lớp 12	Dự đoán thông qua thông tin người dùng nhập.
		Dự đoán thông qua file Excel do người dùng upload.
	Dự đoán từ bài toán thi THPT	Dự đoán thông qua thông tin người dùng nhập.
		Dự đoán thông qua file Excel do người dùng upload.

	Tải kết quả	Tải kết quả dự đoán từ 3 model dưới dạng file Excel để đánh giá và tham khảo.
	Theo dõi số lần submit	Hiển thị số lần submit trong ngày và tổng số lần submit từ hệ thống.
ADMIN	Đăng nhập	Đăng nhập với tài khoản đã đăng ký để truy cập các chức năng quản trị.
	Hiển thị hiệu suất	Xem thông tin về latency, mức sử dụng CPU, RAM, số dự đoán/giây, và hiệu suất tổng thể.
	Train lại model	Thêm dữ liệu mới để train lại model.
	Đánh giá models	Hiển thị độ chính xác của model sau khi train (MAE, MAPE, MSE, RMSE, Accuracy).
	Chấp nhận hoặc từ chối model mới	Quyết định sử dụng model mới hoặc giữ lại model cũ.
	Cập nhật dữ liệu	Cập nhật lại bộ dữ liệu đã train nếu sử dụng model mới.
	Trực quan hóa dữ liệu	Vẽ lại biểu đồ để thể hiện dữ liệu đã được cập nhật.

CHƯƠNG IV: KẾT QUẢ THỰC HIỆN

1. Trang web dành cho người dùng:

1.1. Giao diện nhập liệu:

Người dùng có thể lựa chọn trang để thực hiện dự đoán điểm trung bình năm học lớp 12 hoặc điểm thi tốt nghiệp thông qua 2 hình thức nhập liệu là nhập trực tiếp hoặc nhập bằng file excel.

The screenshot shows the website interface for the 12th Grade Average Score Prediction. The header is blue with the UIT logo and the text "Dự đoán điểm trung bình lớp 12". Below the header, there are two tabs: "Điểm trung bình 12" (selected) and "Điểm thi Tốt Nghiệp". The main content area has a red title "Dự đoán điểm trung bình lớp 12". Below the title, there are two radio buttons: "Nhập vào ô" (selected) and "Nhập bằng Excel". Below the radio buttons, there is a table with 5 rows and 4 columns. The columns are labeled "HK1 Lớp 10", "HK2 Lớp 10", "HK1 Lớp 11", and "HK2 Lớp 11". The rows are labeled "Điểm Toán", "Điểm Văn", "Điểm Lý", "Điểm Hóa", and "Điểm Sinh". Each cell in the table contains a text input field. To the right of the table, there is a box with the text "Tổng số lần submit: 1" and "Số lần submit hôm nay: 1".

	HK1 Lớp 10	HK2 Lớp 10	HK1 Lớp 11	HK2 Lớp 11
Điểm Toán	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Điểm Văn	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Điểm Lý	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Điểm Hóa	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Điểm Sinh	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Tổng số lần submit: 1
Số lần submit hôm nay: 1

Hình 4: Giao diện nhập liệu trực tiếp.

Hình 5: Giao diện nhập liệu bằng file excel

1.2. Giao diện hiển thị dữ liệu:

Điểm sau khi được trả về từ 3 models sẽ được hiển thị thông qua trang kết quả của trang web. Trường hợp dự đoán thông qua file excel người dùng có thể di chuyển qua lại giữa các kết quả, đồng thời tải xuống kết quả thành file excel.

Hình 6: Giao diện hiển thị kết quả dự đoán trực tiếp.

UIT
Dự đoán kết quả thi THPT

Điểm trung bình 12

Điểm thi Tốt Nghiệp

Kết quả dự đoán điểm thi THPT

« Previous 197 trần thị ngọc ý Next » Download

Linear Regression

Điểm học thi tốt nghiệp

Điểm Toán	Điểm Văn	Điểm Lý	Điểm Hóa	Điểm Sinh	Điểm Anh
8.1	7	6.7	7.2	5.5	5.1

Trở về

Multilayer Perceptron

Tổng số lần submit: 2
Số lần submit hôm nay: 2

Hình 7: Giao diện hiển thị kết quả dự đoán bằng excel.

1.3. Giao diện mô tả dữ liệu:

Người dùng có thể xem các biểu đồ thể hiện, so sánh độ chính xác, biểu đồ thể hiện phổ điểm, phân loại học sinh theo học lực, điểm số, ma trận tương quan,... được xây dựng từ các dữ liệu được train cho models và các dữ liệu được models dự đoán.



Hình 8: Giao diện mô tả dữ liệu.

2. Trang web dành cho admin:

2.1. Đăng nhập, đổi mật khẩu:

Admin cần đăng nhập để có thể sử dụng các route quan trọng. Tài khoản này chỉ lưu hành nội bộ nhóm phát triển và được thay đổi thường xuyên nhằm đảm bảo tính an toàn cho hệ thống.

Hình 9: Giao diện đăng nhập.

Hình 10: Giao diện đổi mật khẩu.

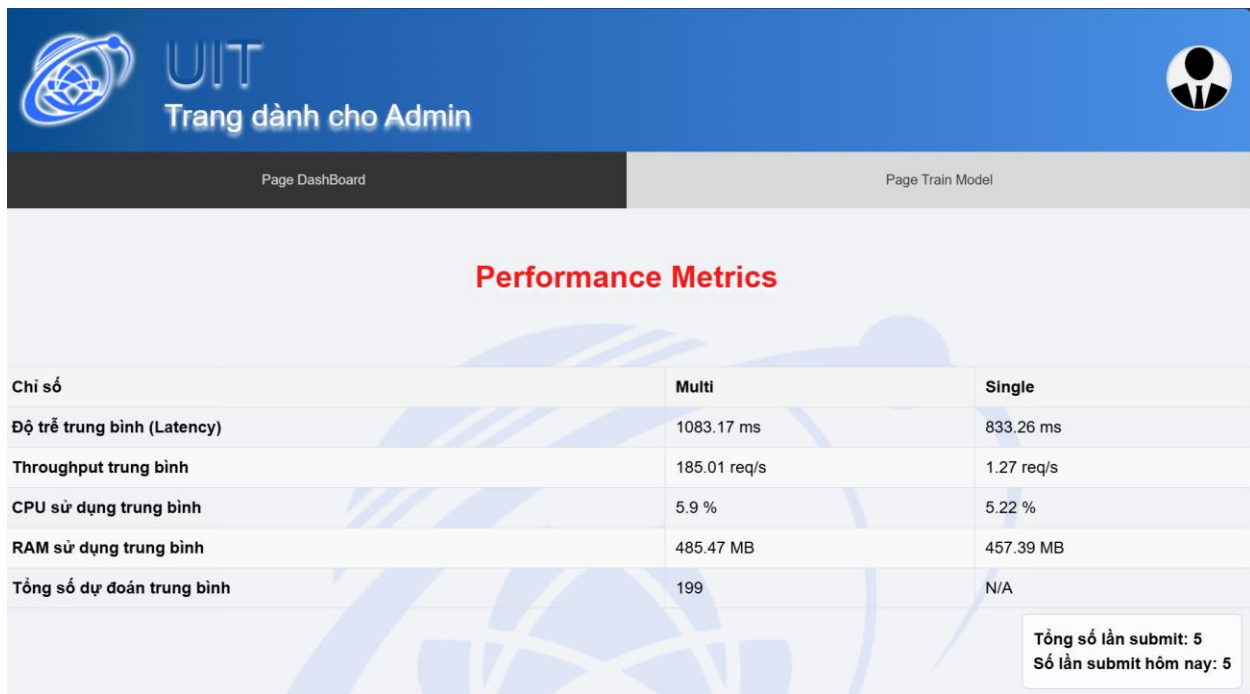
2.2. Giao diện dashboard:

Mỗi lần thực hiện tính toán ứng dụng đồng thời thực hiện đo đạt các thông số hiệu suất của ứng dụng và lưu trữ vào database nhằm giúp người quản trị dễ dàng đánh giá hiệu suất cũng như độ ổn định của ứng dụng web.

Các thông số bao gồm:

- Độ trễ: thời gian phản hồi từ lúc người dùng chọn dự đoán đến lúc trả về kết quả.
- Throughput: số lần dự đoán được thực hiện trong 1 giây.
- CPU sử dụng: Mức độ sử dụng CPU trong 1 lần dự đoán
- Ram sử dụng: Mức độ sử dụng Ram trong 1 lần dự đoán.
- Tổng số dự đoán: là số lần dự đoán trên 1 lần request từ người dùng.

Các thông số trên được lưu trữ và tính toán trung bình và hiển thị trên trang theo dõi hoạt động của Admin.



Hình 11: Giao diện hiển thị hiệu suất.

2.3. Giao diện cho admin nhập dữ liệu cho quá trình train model:

UIT
Trang dành cho Admin

Page DashBoard Page Train Model

Admin Training

Traning Model 12 ☒ Training Model TN_TN ☐ Training Model TN_XH ☐

Excel File

Excel Input: No file chosen

Tổng số lần submit: 5
Số lần submit hôm nay: 5

Hình 12: Giao diện nhập liệu cho quá trình train model.

Admin có thể lựa chọn training cho loại mô hình nào và lựa chọn dữ liệu phù hợp.

2.4. Giao diện cho admin xem các thông số về độ tin cậy của model trước và sau khi train:

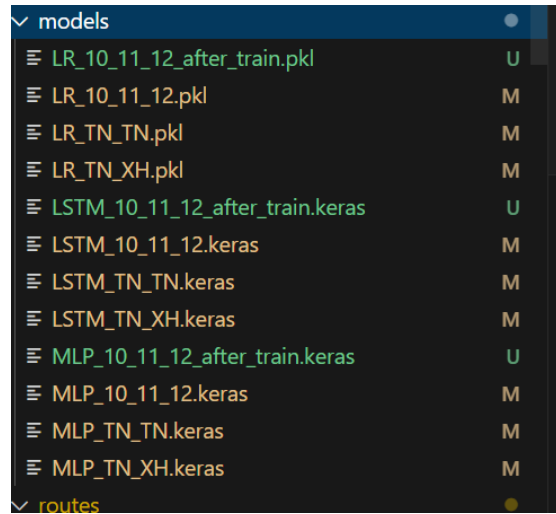


Mô hình	Chỉ số	Trước khi huấn luyện	Sau khi huấn luyện
Linear Regression	MAE	0.93	0.42
	MAPE (%)	11.69	5.62
	MSE	1.37	0.31
	RMSE	1.08	0.54
	Accuracy (%)	88.31%	94.38%
MLP	MAE	0.97	0.5
	MAPE (%)	12.16	6.82
	MSE	1.48	0.44

Tổng số lần submit: 5
 Số lần submit hôm nay: 5

Hình 13: Giao diện so sánh độ tin cậy của model.

Sau khi train 2 phiên bản model được tạo ra ứng dụng thực hiện tính toán độ chính xác của 2 phiên bản trên cùng bộ dữ liệu test và hiển thị bảng so sánh lên trang web. Người dùng quản trị có thể so sánh, đánh giá và đưa ra quyết định có chấp nhận bộ dữ liệu trên hay không.



Hình 14: Giao diện mô tả kết quả sau khi train.

Nếu chấp nhận model sau khi train và bộ dữ liệu sau khi được cập nhật sẽ được ghi đè lên model và bộ dữ liệu cũ. Nếu từ chối toàn bộ kết quả sẽ bị xóa.

CHƯƠNG V: KẾT LUẬN

1. Kết quả đạt được

Ứng dụng **EduPredictOps** hoạt động khá tốt và hiệu quả đáp ứng đầy đủ nhu cầu sử dụng từ phía người dùng, hệ thống tương đối dễ sử dụng. Về phía người quản trị các thông số giúp ích khá nhiều trong việc đưa ra đánh giá và giải pháp cho ứng dụng.

Về mặt giao diện tương đối dễ sử dụng người dùng có thể làm quen ngay mà không gặp bất cứ khó khăn gì.

2. Hạn chế:

Nhóm gặp nhiều khó khăn khi xây dựng phân quyền trên blueprint cũng như thực hiện bảo vệ database. Quá trình train model cũng như chấp nhận model và cập nhật lại bộ dữ liệu cần nguồn tài nguyên khá lớn. Dẫn đến còn nhiều hạn chế khi đưa vào thực tiễn.

3. Hướng phát triển dự án:

- Nâng cấp giao diện thẩm mỹ và dễ sử dụng hơn: Cải thiện trải nghiệm người dùng (UX/UI) bằng cách thiết kế giao diện hiện đại, trực quan và phù hợp với đa dạng đối tượng người dùng.
- Tích hợp các công cụ hỗ trợ quản lý vòng đời Model Machine Learning như: MLFlow hoặc các công cụ tương tự để quản lý, theo dõi và triển khai các phiên bản model một cách hiệu quả.
- Tích hợp CI/CD cho ứng dụng web: tự động hóa quy trình kiểm tra, xây dựng, và triển khai ứng dụng web, tối ưu hóa quy trình cập nhật model và giao diện web mà không làm gián đoạn hoạt động của người dùng.
- Nâng cấp bộ dữ liệu: Xây dựng hệ thống tự động hóa quá trình làm sạch và tiền xử lý dữ liệu (xử lý giá trị thiếu, chuẩn hóa dữ liệu, phát hiện ngoại lệ).
- Mở rộng quy mô ứng dụng.

Tài liệu tham khảo

[Linear Regression \(Python Implementation\) - GeeksforGeeks](#) (1)

[Multi-Layer Perceptron Learning in Tensorflow - GeeksforGeeks](#)(2)

[Multi-Layer Perceptron Learning in Tensorflow - GeeksforGeeks](#)(3)