

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



Khai Thác Dữ Liệu Và Ứng Dụng – CS313.N22

Đề tài

**Dự đoán kết quả học tập của các môn học mà sinh viên lựa
chọn cho học kỳ tiếp theo**

GVHD: Nguyễn Thị Anh Thư

Nhóm 3:

Bùi Nguyễn Anh Trung (NT)	20520332
Hồ Thanh Tịnh	20520813
Nguyễn Trần Minh Anh	20520394
Lê Nguyễn Bảo Hân	20520174
Nguyễn Văn Đức Ngọc	20521666

Hồ Chí Minh, 28 tháng 05 năm 2023

MỤC LỤC

1. Tổng quan đề tài	1
Giới thiệu.....	1
Thách thức của bài toán	1
Đối tượng và phạm vi.....	2
Mục tiêu.....	2
2. Nội dung thực hiện	3
2.1. Các bước tiền xử lý dữ liệu	3
2.1.1. Tiền xử lý dữ liệu cho phương pháp Recommendation System	3
2.1.1.1. Bảng sinh viên	3
2.1.1.2. Bảng môn học	3
2.1.1.3. Bảng điểm.....	4
2.1.2. Tiền xử lý dữ liệu cho phương pháp Machine Learning.....	5
2.1.3. Tiền xử lý dữ liệu cho phương pháp Neural Network	5
2.2. Các thuộc tính sử dụng.....	7
2.2.1. Các thuộc tính sử dụng cho phương pháp Recommendation System.....	7
2.2.2. Các thuộc tính sử dụng cho phương pháp Machine Learning	11
2.2.3. Các thuộc tính sử dụng cho phương pháp Neural Network.....	13
2.3. Phương pháp đề xuất.....	16
2.3.1. Hướng tiếp cận Machine Learning	16
2.3.1.1. Linear Regression và các biến thể	16
Giới thiệu mô hình.....	16
Ordinary Least Squares (OLS) Regression.....	17
Lasso Regression	18
Ridge regression	18

2.3.1.2. Decision Tree Regression	18
Giới thiệu mô hình	18
Các tham số đầu vào để cấu hình mô hình	20
2.3.1.3. Xgboost Regression	20
Giới thiệu mô hình	20
Các tham số đầu vào để cấu hình mô hình	21
2.3.1.4. Nhận xét.....	21
2.3.2. Hướng tiếp cận Neural Network	22
Tổng quan thuật toán	22
Các lớp trong mô hình	23
Cách thức hoạt động	25
2.3.3. Hướng tiếp cận hệ khuyến nghị (Recommendation System).....	25
2.3.3.1. Hệ khuyến nghị (Recommendation System)	25
Định nghĩa	25
Các mô hình hệ khuyến nghị	25
Phân tích các bước thực hiện	26
2.3.3.2. Collaborative Filtering	27
Định nghĩa	27
Các hướng tiếp cận	28
2.3.3.3. Content Filtering.....	30
Phương pháp 1: Đề xuất điểm dựa trên điểm các môn học tương tự mà sinh viên đó đã học	31
Phương pháp 2: Đề xuất điểm dựa trên điểm các sinh viên tương tự đã học qua môn học đó.....	31
Phương pháp tính điểm dựa trên độ tương đồng	31
3. Cài đặt thực nghiệm	32

3.1. Dataset.....	32
3.2. Phương pháp đánh giá.....	32
3.3. Các phương pháp thực nghiệm.....	32
3.4. Kết quả thực nghiệm	34
4. Kết luận và hướng phát triển	34
5. Tài liệu tham khảo	35
6. Bảng phân công công việc.....	36



1. Tổng quan đề tài

Giới thiệu

Phương thức đào tạo theo tín chỉ đem lại sự linh hoạt trong nhiều hoàn cảnh nhưng đồng thời đặt ra nhiều thách thức cho cả nhà trường và sinh viên. Vấn đề xây dựng lộ trình học không phù hợp sẽ gây ảnh hưởng đến hiệu quả học tập và tỷ lệ tốt nghiệp đúng hạn. Vì vậy, việc phát triển mô hình dự đoán điểm các môn học mà sinh viên lựa chọn trong học kỳ tiếp theo là một nhu cầu cần thiết.

Với dữ liệu lịch sử bảng điểm cùng với các thông tin liên quan về sinh viên và môn học, bài toán dự đoán kết quả học tập này có thể được thực hiện bằng nhiều phương pháp. Báo cáo đề tài này tập trung xây dựng trên ba hướng tiếp cận: mô hình Máy học, mô hình Neural Network và mô hình hệ thống khuyến nghị (Recommendation System – RS), Phát biểu bài toán

Áp dụng các mô hình hồi quy, mô hình neural network và các kỹ thuật của hệ thống khuyến nghị để dự đoán kết quả môn học mà sinh viên lựa chọn cho học kỳ tiếp theo.

Input: Đầu vào của bài toán bao gồm các thông tin về học kỳ, thông tin về sinh viên và thông tin môn học. Trong mỗi học kỳ, một sinh viên có thể học nhiều môn. Bảng dữ liệu cung cấp danh sách các môn học mà sinh viên đã đăng ký, điểm số tương ứng trong các học kỳ trước đó, và các đặc điểm của môn học.

Output: Đầu ra của bài toán là điểm dự đoán sinh viên sẽ đạt được ở các môn học trong học kỳ tiếp theo.

Thách thức của bài toán

1. Xử lý dữ liệu: Dữ liệu về sinh viên, môn học và điểm có thể thiếu sót, chứa nhiều nhiễu và giá trị bất thường do nhiều yếu tố khác nhau. Mô hình có thể gặp khó khăn trong việc học các mẫu phức tạp. Việc xử lý dữ liệu thiếu sót nhưng vẫn đảm bảo tính chính xác của dữ liệu là một thách thức trong đề tài này.

2. Sự phụ thuộc vào ngữ cảnh: Điểm môn học có thể phụ thuộc vào nhiều yếu tố ngữ cảnh, cần tìm ra phương hướng đánh giá một cách chính xác và khách quan nhất. Những đánh giá đòi hỏi được xem xét trên nhiều khía cạnh, nhiều loại đối tượng sinh viên và nhiều môn học khác nhau. Cần đưa ra hướng xử lý cho sự phụ thuộc vào môn học (sinh viên có sở trường thiên về một môn học nào đó) và yêu cầu của môn học (môn học có độ khó khác nhau).

Đối tượng và phạm vi

Đối tượng của đề tài là các sinh viên trong một hệ thống giáo dục hoặc tổ chức giảng dạy cụ thể, có một khối lượng dữ liệu nhất định, trong đó có thông tin về điểm môn học của các học kỳ trước.

Phạm vi của đề tài bao gồm việc xây dựng một mô hình dự đoán để ước lượng điểm môn học học kỳ tiếp theo dựa trên thông tin lịch sử điểm môn học của sinh viên. Đồng thời, phạm vi cũng bao gồm việc đánh giá kết quả dự đoán và hiểu rõ các yếu tố ảnh hưởng đến kết quả dự đoán, nhằm cung cấp thông tin hữu ích cho quyết định giáo dục và hỗ trợ cho quá trình học tập của sinh viên.

Đề tài chỉ xây dựng mô hình dự đoán kết quả của môn học nên không chứa những thông tin về cách thức quản lý chương trình đào tạo như ràng buộc môn học bắt buộc, môn học tiên quyết, môn học tự chọn/nhóm môn học tự chọn,...

Không xét đến yếu tố phát sinh môn học mới trong chương trình đào tạo.

Mục tiêu

1. Nghiên cứu, khảo sát và áp dụng các kỹ thuật của mô hình máy học, mô hình học sâu và hệ thống gợi ý cho nhiệm vụ dự đoán kết quả học tập các môn học của sinh viên.
2. Xác định ưu và nhược điểm, cũng như sự phối hợp giữa các phương pháp được đề xuất. Từ đó, đánh giá giải thuật và so sánh với các phương pháp khác.
3. Sử dụng các trọng số cho các đặc trưng khác nhau để khám phá các đặc điểm của sinh viên dựa trên kết quả các môn học.

2. Nội dung thực hiện

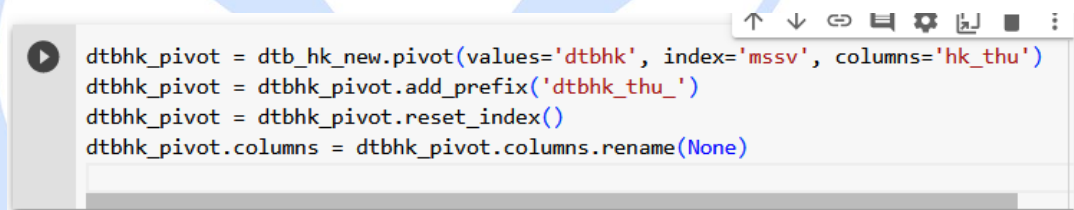
2.1. Các bước tiền xử lý dữ liệu

2.1.1. Tiền xử lý dữ liệu cho phương pháp Recommendation System

2.1.1.1. Bảng sinhvien

Các bước tiền xử lý để tạo nên bảng sinh viên:

- Load dữ liệu: đầu tiên, load bộ dữ liệu 01.sinhvien và diem_Thu vào cấu trúc dữ liệu DataFrame của thư viện pandas.
- One-hot encoding: sử dụng cột 'khoa' có trong bảng 01.sinhvien để tạo nên các cột tương ứng với từng giá trị khác nhau của 'khoa', sau đó tích hợp các cột này vào bảng diem_Thu bằng 'mssv'.
- Tính điểm trung bình học kỳ: trước tiên tạo cột 'hk_thu' thống kê số học kỳ sinh viên đã theo học tại trường bằng cách nhóm dữ liệu bằng 'mssv', sau đó dùng hàm cumcount() để tính số học kỳ. Kế đó tạo pivot table với index là 'mssv', columns là 'hk_thu' và values là 'dtbhk' load từ bảng sinhvien_dtb_hocky. Làm tương tự với số tín chỉ học kỳ.



```
dtbhk_pivot = dtb_hk_new.pivot(values='dtbhk', index='mssv', columns='hk_thu')
dtbhk_pivot = dtbhk_pivot.add_prefix('dtbhk_thu_')
dtbhk_pivot = dtbhk_pivot.reset_index()
dtbhk_pivot.columns = dtbhk_pivot.columns.rename(None)
```

Hình 2.1 Minh họa cho cách tạo pivot table từ 'mssv', 'hk_thu' và 'dtbhk'

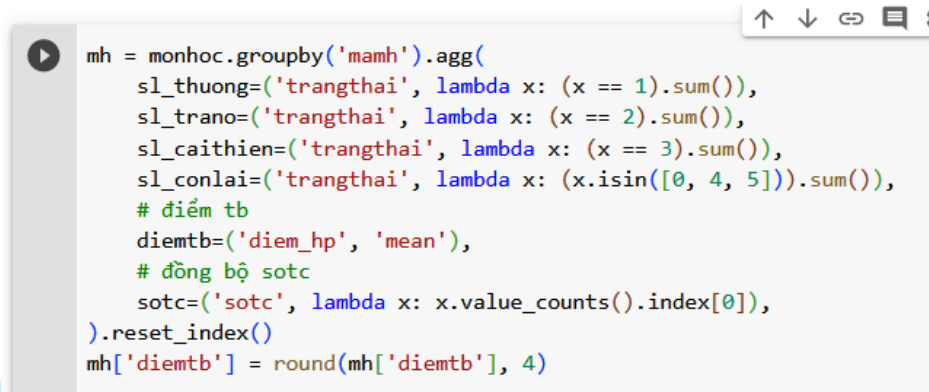
- Gộp dữ liệu: gộp các cột khoa, điểm trung bình học kỳ, số tín chỉ học kỳ vào bảng diem_Thu đã được nhóm bằng 'mssv'.

Ta có được bảng sinhvien với 17881 mẫu dữ liệu có thông tin về khoa, điểm và số tín chỉ học kỳ.

2.1.1.2. Bảng monhoc

Các bước xử lý để tạo nên bảng monhoc gồm có:

- Load dữ liệu: load bộ diem_Thu dưới dạng cấu trúc DataFrame của pandas.
- Tạo các cột thống kê số lượng đăng ký môn bình thường, để trả nợ hoặc để cải thiện: đầu tiên nhóm dữ liệu theo 'mamh', sau đó dùng hàm sum() lên các giá trị của trạng thái để thống kê số lượng.



```

mh = monhoc.groupby('mamh').agg(
    sl_thuong=('trangthai', lambda x: (x == 1).sum()),
    sl_trano=('trangthai', lambda x: (x == 2).sum()),
    sl_caithien=('trangthai', lambda x: (x == 3).sum()),
    sl_conlai=('trangthai', lambda x: (x.isin([0, 4, 5])).sum()),
    # điểm tb
    diemtb=('diem_hp', 'mean'),
    # đồng bộ sotc
    sotc=('sotc', lambda x: x.value_counts().index[0]),
).reset_index()
mh['diemtb'] = round(mh['diemtb'], 4)

```

Hình 2.2 Minh họa cho thao tác thống kê số lượng theo giá trị 'trangthai'

- Tạo cột điểm trung bình: nhóm dữ liệu theo 'mamh', sau đó dùng hàm mean để tính điểm trung bình từng môn.
- One-hot encoding lên cột môn khoa: sau khi phân chia khoa quan theo tên môn học, dùng one-hot encode để biểu diễn các giá trị khác nhau của 'monkhoa'.
- Tạo các cột thống kê số lượng sinh viên đăng ký môn: tạo cột 'hk_thu' để biểu diễn số kỳ học của sinh viên như đã làm ở bảng sinhvien, sau đó tạo pivot table với index là 'mamh', columns là 'hk_thu', dùng hàm size để thống kê số lượng sinh viên đăng ký theo từng học kỳ.

2.1.1.3. Bảng diem

- Load dữ liệu diem_Thu thành cấu trúc DataFrame.
- Xoá dữ liệu: xoá các dữ liệu trống ở cột 'diem_hp', các giá trị 0, 4, 5 ở cột 'trangthai' chỉ các trường hợp miễn, hoãn và huỷ môn.
- Tính điểm trung bình bằng cách nhóm dữ liệu theo 'mssv' và 'mamh' sau đó dùng hàm mean.

2.1.2. Tiền xử lý dữ liệu cho phương pháp Machine Learning

Các thông tin cần thiết huấn luyện mô hình máy học cũng tương tự như Recommendation System, dựa trên file `diem_Thu`, gồm thông tin sinh viên, môn học và điểm để dự đoán. Các bước tiền xử lý để tạo nên dữ liệu bao gồm:

- Đọc dữ liệu `diem_Thu` bằng thư viện `pandas`.
- Giữ các thông tin cơ bản về điểm như `'mssv'`, `'mamh'`, `'sotc'`, `'hocky'`, `'namhoc'`, `'trangthai'` và `'diem_hp'` có sẵn trong `'diem_Thu'`.
- Xử lý dữ liệu không hợp lệ: xóa các dữ liệu `NULL` có trong cột `'diem_hp'` để làm sạch dữ liệu.
- Xử lý giá trị ngoại lệ: các giá trị 0, 4, 5 ở cột `'trangthai'` sẽ bị xóa khỏi bảng dữ liệu nhằm xóa đi các mẫu dữ liệu mang thông tin về sự can thiệp của môi trường vào điểm học phần của sinh viên.
- Các thông tin về sinh viên như khoa, giới tính, hệ đào tạo, tình trạng sinh viên và khóa học được tích hợp từ file `01.sinhvien` thông qua `'mssv'`.
- Biến đổi dữ liệu: kèm theo các thông tin cơ bản như trên, sinh viên còn có thông tin về số tín chỉ kỳ học, được thống kê bằng cách nhóm dữ liệu từ file `diem_Thu` lại bằng `'mssv'`, `'hocky'`, `'namhoc'` và tính tổng trên cột `'sotc'`.
- Tính điểm trung bình theo kỳ học: sinh viên sẽ có thông tin điểm trung bình theo từng kỳ học như đã làm ở bảng `sinhvien` của phương pháp Recommendation System, tạo cột `'hk_thu'` để chỉ kỳ học hiện tại của sinh viên, sau đó nhóm dữ liệu theo `'mssv'` và tính trung bình trên `'diem_hp'` để có được giá trị điểm trung bình theo kỳ học.

2.1.3. Tiền xử lý dữ liệu cho phương pháp Neural Network

Tương tự như phương pháp học máy, phương pháp dùng mạng Neural cũng yêu cầu dữ liệu được tổ chức với các thông tin về điểm, sinh viên, môn học được tích hợp thành một bảng dữ liệu, với cột `'diem_hp'` là label và các cột còn lại là dữ liệu đầu vào để huấn luyện mô hình.

- Cũng như các bảng dữ liệu trên, để tạo dữ liệu cho Neural Network trước hết cần tạo DataFrame dựa trên đọc dữ liệu diem_Thu.
- Sau đó tiến hành xóa các dữ liệu không cần thiết như các giá trị NULL ở cột 'diem_hp' và các giá trị trạng thái 0, 4, 5 để giữ nguyên giá trị thông tin cần thiết ở dữ liệu.
- Tạo bảng dữ liệu với các cột 'mssv', 'mamh', 'sotc', 'hocky', 'namhoc', 'trangthai', 'diem_hp' để mô hình có thể có được thông tin cơ bản về từng mẫu dữ liệu.
- Tích hợp các thuộc tính về sinh viên ở bảng 01.sinhvien, 05.ThiSinh, 10.diemrl và sinhvien_dtb_hocky như 'khoa', 'hedt', 'khoahoc', 'dien_tt', 'diem_tt', 'drl' và 'sotchk' để có được nhưng thông tin cơ bản về hiện tại cũng thông tin quá khứ của sinh viên.
- Tạo ra các cột điểm trung bình của sinh viên theo học kỳ nhằm theo dõi tiến độ học vấn của sinh viên bằng cách tạo pivot table với index là 'mssv', columns là 'namhoc' và 'hocky', và nhập các giá trị điểm trung bình, ta có bảng chứa thông tin điểm trung bình của sinh viên theo từng học kỳ.

```
diem_dtb = diem[['mssv', 'mamh', 'hocky', 'namhoc', 'sotc', 'diem_hp']]
diem_dtb['dtb'] = diem_dtb.groupby(['mssv', 'namhoc', 'hocky'], sort=False)['sotc'].transform('sum')

for i in range(0, 656349, 1):
    diem_dtb.iat[i, 6] = (diem_dtb.iat[i, 5] * diem_dtb.iat[i, 4]) / diem_dtb.iat[i, 6]

diem_dtb['dtb'] = diem_dtb.groupby(['mssv', 'namhoc', 'hocky'], sort=False)['dtb'].transform('sum')

diem_tb_pivot = diem_dtb.pivot_table(index='mssv', columns=['namhoc', 'hocky'], values=['dtb'], fill_value=0)
diem_tb_pivot = diem_tb_pivot.reset_index()
```

Hình 2.3 Minh họa cho cách tính điểm trung bình theo công thức tích lũy

- Tạo ra cột điểm trung bình môn học để xem xét sự thay đổi của môn học theo từng học kỳ bằng cách nhóm dữ liệu tương tự như ở điểm trung bình sinh viên, nhưng lần này là với giá trị của 'diem_hp' đã được áp dụng hàm mean.
- Tạo ra bảng thống kê số lượng sinh viên đăng ký môn học bình thường, đăng ký để trả nợ và đăng ký để cải thiện nhằm cung cấp thêm cho mô hình tỉ lệ đăng ký môn và tỉ lệ qua môn của môn học theo từng học kỳ. Bằng cách tạo ra các cột 'thuong', 'trano', 'caithien' riêng ghi nhận số lượng đăng ký môn

ở các trạng thái này, sau đó nhóm dữ liệu theo ‘mamh’, ‘namhoc’, ‘hocky’ và dùng hàm sum để tính tổng số lượng.

Cuối cùng ta có được bảng dữ liệu với thông tin cần thiết để huấn luyện mô hình neural.

Mỗi dữ liệu đều có số lượng thuộc tính riêng biệt nhưng nhìn chung, ở mỗi bộ đều cần mã hoá các thông tin quan trọng như ‘mssv’ và ‘mamh’ để mô hình có thể huấn luyện trên bộ dữ liệu.

```
from sklearn.preprocessing import LabelEncoder

mssv_encoder = LabelEncoder()
mamh_encoder = LabelEncoder()
```

Hình 2.4 Minh hoạ cho cách mã hoá dữ liệu từ dạng string sang int

Hoàn tất mã hoá, ta đã có được bộ dữ liệu được xây dựng và xử lý hoàn chỉnh, để bắt đầu huấn luyện ta tiến hành phân chia tỷ lệ trên dữ liệu cho tập huấn luyện và tập thực nghiệm. Với tập thực nghiệm là các năm thuộc về năm 2022 và tập huấn luyện là dữ liệu của các năm trở về trước, đồng thời tạo nên tập train và test.

```
train_data = data_filled.loc[data['namhoc'] < 2022]
test_data = data_filled.loc[data['namhoc'] == 2022]

X_train = train_data.drop(['diem_hp'], axis=1)
y_train = train_data['diem_hp']

X_test = test_data.drop(['diem_hp'], axis=1)
y_test = test_data['diem_hp']
```

Hình 2.5 Minh hoạ cho việc phân chia dữ liệu thành tập train và test

2.2. Các thuộc tính sử dụng

2.2.1. Các thuộc tính sử dụng cho phương pháp Recommendation System

Bảng 2.1 Mô tả bảng SINHVIEN và các thuộc tính

Bảng SINHVIEN				
STT	Thuộc tính	Nội dung	Kiểu dữ liệu	Diễn giải
1	mssv	Mã sinh viên	Chuỗi	Chuỗi 40 ký tự

2	CNPM	Sinh viên có thuộc khoa Công nghệ phần mềm hay không	Số	Gồm các trạng thái: • 0: không • 1: có
3	HTTT	Sinh viên có thuộc khoa Hệ thống thông tin hay không	Số	Gồm các trạng thái: • 0: không • 1: có
4	KHMT	Sinh viên có thuộc khoa Khoa học máy tính hay không	Số	Gồm các trạng thái: • 0: không • 1: có
5	KTMT	Sinh viên có thuộc khoa Kỹ thuật máy tính hay không	Số	Gồm các trạng thái: • 0: không • 1: có
6	KTTT	Sinh viên có thuộc khoa Khoa học và kỹ thuật thông tin hay không	Số	Gồm các trạng thái: • 0: không • 1: có
7	MTT&TT	Sinh viên có thuộc khoa Mạng máy tính và truyền thông hay không	Số	Gồm các trạng thái: • 0: không • 1: có
8	dtbhc_thu_x	Điểm trung bình học kỳ	Số thực	Với x là thứ tự học kỳ của sinh viên, x chạy từ 0 tới 27
11	sotchk_thu_x	Số tín chỉ học kỳ	Số	Với x là thứ tự học kỳ của sinh viên, x chạy từ 0 tới 27
12	dtbtl	Điểm trung bình tích lũy	Số thực	Từ 0 đến 10

Bảng 2.2 Mô tả bảng MONHOC và các thuộc tính

Bảng MONHOC				
STT	Thuộc tính	Nội dung	Kiểu dữ liệu	Diễn giải

1	mamh	Mã môn học	Chuỗi	Biểu diễn dưới dạng ký hiệu và số như 'IT001', 'ADENG1',...
2	sl_thuong	Số lượng sinh viên đăng ký môn học bình thường	Số	
3	sl_trano	Số lượng sinh viên đăng ký môn học để trả nợ	Số	
4	sl_caithien	Số lượng sinh viên đăng ký môn học để cải thiện	Số	
5	diemtb	Điểm trung bình của môn học	Số thực	<ul style="list-style-type: none"> • Điểm trung bình của tất cả sinh viên đã đăng ký môn học. • Từ 0 đến 10
6	sotc	Số tín chỉ của môn học	Số	<ul style="list-style-type: none"> • Từ 0 đến 5 • sotc = 0 là những môn không tính vào ĐTB và điểm tích lũy
7	monchung	Có phải môn chung hay không	Số	<p>Gồm các trạng thái:</p> <ul style="list-style-type: none"> • 0: không • 1: có
8	KHMT	Có phải môn do khoa khoa Khoa học máy tính quản lý không	Số	<p>Gồm các trạng thái:</p> <ul style="list-style-type: none"> • 0: không • 1: có
9	CNPM	Có phải môn do khoa khoa Công nghệ phần mềm quản lý không	Số	<p>Gồm các trạng thái:</p> <ul style="list-style-type: none"> • 0: không • 1: có
10	HTTT	Có phải môn do khoa khoa Hệ thống thông tin quản lý không	Số	<p>Gồm các trạng thái:</p> <ul style="list-style-type: none"> • 0: không • 1: có

11	MTT	Có phải môn do khoa Mạng máy tính và truyền thông quản lý không	Số	Gồm các trạng thái: <ul style="list-style-type: none"> • 0: không • 1: có
12	KTMT	Có phải môn do khoa khoa Kỹ thuật máy tính quản lý không	Số	Gồm các trạng thái: <ul style="list-style-type: none"> • 0: không • 1: có
13	KTTT	Có phải môn do khoa khoa Khoa học và kỹ thuật thông tin quản lý không	Số	Gồm các trạng thái: <ul style="list-style-type: none"> • 0: không • 1: có
14	slsruhk_thu_x	Số lượng sinh viên đăng ký môn học theo từng học kỳ.	Số	x chạy từ 0 tới 27

Bảng 2.3 Mô tả bảng DIEM và các thuộc tính

Bảng DIEM				
STT	Thuộc tính	Nội dung	Kiểu dữ liệu	Diễn giải
1	mssv	Mã sinh viên	Chuỗi	Chuỗi 40 ký tự
2	mamh	Mã môn học	Chuỗi	Biểu diễn dưới dạng ký hiệu và số như 'IT001', 'ADENG1',...
3	mean_diem_hp	Điểm học phần trung bình	Số thực	<ul style="list-style-type: none"> • Tính trung bình vì có trường hợp sinh viên học lại một môn nhiều lần • Từ 0 đến 10

2.2.2. Các thuộc tính sử dụng cho phương pháp Machine Learning

Bảng 2.4 Mô tả bảng ML_DATA và các thuộc tính

Bảng ML_DATA				
STT	Thuộc tính	Nội dung	Kiểu dữ liệu	Diễn giải
1	mssv	Mã sinh viên	Chuỗi	Chuỗi 40 ký tự
2	mamh	Mã môn học	Chuỗi	Biểu diễn dưới dạng ký hiệu và số như 'IT001', 'ADENG1',...
3	sotc	Số tín chỉ môn học	Số	<ul style="list-style-type: none"> Từ 0 đến 5 sotc = 0 là những môn không tính vào ĐTB và điểm tích lũy
4	hocky	Học kỳ	Số	Học kỳ 1, 2 và 3
5	namhoc	Năm học	Số	Từ năm 2006 tới năm 2022
6	diem_hp	Điểm học phần của môn	Số thực	Từ 0 đến 10
7	trangthai	Trạng thái đăng ký học phần	Số	Gồm các trạng thái: <ul style="list-style-type: none"> 0: Hủy 1: Bình Thường 2: Trả Nợ 3: Cải Thiện 4: Miễn 5: Hoãn

8	meanmh	Điểm trung bình môn học	Số thực	<ul style="list-style-type: none"> • Là điểm trung bình của tất cả các học sinh học trong học kỳ, năm học đó. • Dùng để thống kê do sẽ bị loại bỏ trong quá trình huấn luyện.
9	sotckihoc	Số tín chỉ học kỳ	Số	Tổng số tín chỉ của sinh viên trong học kỳ, năm học đó.
10	diemtbkihoc	Điểm trung bình học kỳ	Số thực	<ul style="list-style-type: none"> • Điểm trung bình của sinh viên trong học kỳ, năm học đó. • Dùng để thống kê do sẽ bị loại bỏ trong quá trình huấn luyện.
11	khoa	Khoa	Chuỗi	Có 6 khoa: CNPM, HTTT, KHMT, KTTT, MMT&TT, KTMT.
12	gioitinh	Giới tính	Số	Gồm các trạng thái: <ul style="list-style-type: none"> • 0: Nữ • 1: Nam • 2: Chưa biết
13	hedt	Hệ đào tạo	Chuỗi	Có 5 hệ: CLC, CNTN, CQUI, CTTT, KSTN.
14	tinhtinh	Tình trạng	Số	Gồm các tình trạng: <ul style="list-style-type: none"> • 1: • 2: Cảnh cáo • 3: • 4:

				<ul style="list-style-type: none"> • 5: Thôi học • 6: • 7: Gia hạn • 8: Tự do
15	khoa_hoc	Khoá học	Số	Từ 8 đến 14
16	ki_thu	Kỳ học thứ	Số	<ul style="list-style-type: none"> • Thể hiện số kỳ học hiện tại của sinh viên. • Ví dụ: sinh viên khoá 2020, ở học kỳ 2 năm học 2023 đang là kỳ thứ 6.
17	x	Điểm trung bình theo từng học kỳ	Số thực	<ul style="list-style-type: none"> • Điểm trung bình của sinh viên theo từng kỳ học. • x chạy từ 1 tới 16. Tức là sinh viên được ghi nhận tối đa tới 16 kỳ học.

2.2.3. Các thuộc tính sử dụng cho phương pháp Neural Network

Bảng 2.5 Mô tả bảng DL_DATA và các thuộc tính

Bảng DL_DATA				
STT	Thuộc tính	Nội dung	Kiểu dữ liệu	Diễn giải
1	mssv	Mã sinh viên	Chuỗi	Chuỗi 40 ký tự

2	mamh	Mã môn học	Chuỗi	Biểu diễn dưới dạng ký hiệu và số như 'IT001', 'ADENG1',...
3	sotc	Số tín chỉ môn học	Số	<ul style="list-style-type: none"> Từ 0 đến 5 sotc = 0 là những môn không tính vào ĐTB và điểm tích lũy
4	hocky	Học kỳ	Số	Học kỳ 1, 2 và 3
5	namhoc	Năm học	Số	Từ năm 2013 tới năm 2022
6	trangthai	Trạng thái đăng ký học phần	Số	<p>Gồm các trạng thái:</p> <ul style="list-style-type: none"> 0: Hủy 1: Bình Thường 2: Trả Nợ 3: Cải Thiện 4: Miễn 5: Hoãn
7	diem_hp	Điểm học phần của môn	Số thực	Từ 0 đến 10
8	khoa	Khoa	Số	<p>Gồm các trạng thái:</p> <ul style="list-style-type: none"> 0: Công nghệ phần mềm 1: Hệ thống thông tin 2: Khoa học máy tính 3: Kỹ thuật máy tính 4: Khoa học và kỹ thuật thông tin 5: Mạng máy tính và truyền thông 6: Không có thông tin

9	hedt	Hệ đào tạo	Số	<p>Gồm các trạng thái:</p> <ul style="list-style-type: none"> • 0: Chất lượng cao • 1: Cử nhân tài năng • 2: Chính quy • 3: Chương trình tiên tiến • 4: Kỹ sư tài năng • 5: Không có thông tin
10	khoahoc	Khoá học	Số	Từ 8 đến 14
11	dien_tt	Diện trúng tuyển	Số	<p>Gồm các trạng thái:</p> <ul style="list-style-type: none"> • 0: 3 môn văn hoá • 1: Chứng chỉ quốc tế • 2: Cử tuyển • 3: THPT • 4: Thông tư bộ • 5: Đánh giá năng lực • 6: Ưu tiên bộ • 7: Ưu tiên ĐHQG • 8: Không có thông tin
12	diem_tt	Điểm trúng tuyển	Số thực	Từ 0 đến 1059 hoặc NULL
13	drl	Điểm rèn luyện	Số	<ul style="list-style-type: none"> • Điểm rèn luyện của sinh viên • Từ -30 đến 100
14	sotchk	Số tín chỉ học kỳ	Số	<ul style="list-style-type: none"> • Tổng số tín chỉ trong một học kỳ, năm học của sinh viên • Từ 0 tới 195

15	dtb_x_y	Điểm trung bình theo học kỳ và năm học	Số thực	<ul style="list-style-type: none"> • Điểm trung bình trong một học kỳ của sinh viên • x là học kỳ có giá trị từ 1 tới 3 • y là năm học có giá trị từ 2006 tới 2022 • Từ 0 đến 10 hoặc NULL
16	dtbmon_x_y	Điểm trung bình môn theo học kỳ và năm học	Số thực	<ul style="list-style-type: none"> • Điểm trung bình trong một học kỳ, năm học của môn học • x là học kỳ có giá trị từ 1 tới 3 • y là năm học có giá trị từ 2006 tới 2022 • Từ 0 đến 10 hoặc NULL
17	thuong	Số lượng sinh viên đăng ký môn học bình thường	Số	
18	trano	Số lượng sinh viên đăng ký môn học để trả nợ	Số	
19	caithien	Số lượng sinh viên đăng ký môn học để cải thiện	Số	

2.3. Phương pháp đề xuất

2.3.1. Hướng tiếp cận Machine Learning

2.3.1.1. Linear Regression và các biến thể

Giới thiệu mô hình

Ta có bộ số X có D mẫu dữ liệu, và bộ số y là các giá trị thể hiện lên điểm học phần

của sinh viên.

$$X^{(i)} = (x_1^{(i)}, x_2^{(i)}, x_3^{(i)}, \dots, x_n^{(i)}) \text{ (Bộ giá trị ở dòng thứ } i, \text{ có } n \text{ thuộc tính } x)$$

Ta có hàm biểu diễn quan hệ giữa X và y theo bộ tham số $w = (w_0, w_1, w_2, \dots, w_n)$.

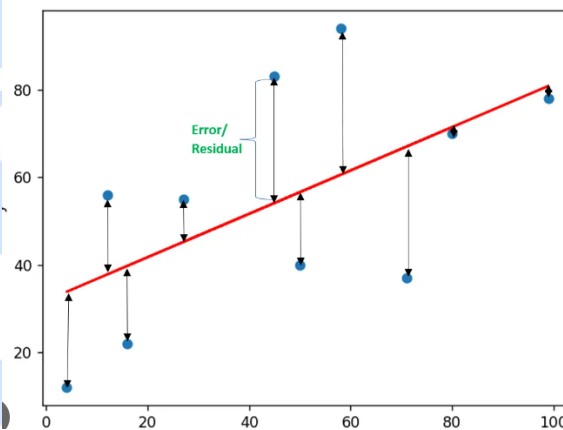
Khác với Linear Regression, giá trị đầu ra y là 1 số thực bất kì nên ta có thể biểu diễn mối quan hệ giữa X, y trong Linear Regression là:

$$Y = w^T X$$

$$Y^{(i)} = w_0 + w_1 * x_1^{(i)} + w_2 * x_2^{(i)} + \dots + w_n * x_n^{(i)}$$

Thực tế là ta khó thể giải hệ phương trình trên 1 cách chính xác được, nên thay vào đó thì ta thường dùng cách là tối ưu hóa các hàm mục tiêu để tìm nghiệm w^T , dựa vào các hàm mục tiêu ta có 1 số mô hình hồi quy như sau:

- Ordinary Least Squares (OLS) Regression
- Lasso Regression
- Ridge Regression



Hình 2.6 Minh họa Linear Regression

Ordinary Least Squares (OLS) Regression

Hàm mục tiêu:

$$w = \min_w \left(\sum_{i=1}^D (y^{(i)} - Y^{(i)})^2 \right)$$

Trong đó:

- $y^{(i)}$ là kết quả thật từ dữ liệu gốc ở cột diem_hp
- $Y^{(i)}$ là kết quả dùng 1 bộ số w nào đó dự đoán

Lasso Regression

Hàm mục tiêu:

$$w = \min_w \left(\sum_{i=1}^D (y^{(i)} - Y^{(i)})^2 + \alpha * ||w||_1 \right)$$

Trong đó:

- $y^{(i)}$ là kết quả thật từ dữ liệu gốc ở cột diem_hp
- $Y^{(i)}$ là kết quả dùng 1 bộ số w nào đó dự đoán
- $||w||_1$ là 1-norm của w

Thông số hiệu chỉnh mô hình: α

Ridge regression

Hàm mục tiêu:

$$w = \min_w \left(\sum_{i=1}^D (y^{(i)} - Y^{(i)})^2 + \alpha * ||w||_2^2 \right)$$

Trong đó $y^{(i)}$ là kết quả thật từ dữ liệu gốc ở cột diem_hp

$Y^{(i)}$ là kết quả dùng 1 bộ số w nào đó dự đoán

$||w||_1$ là 1-norm của w

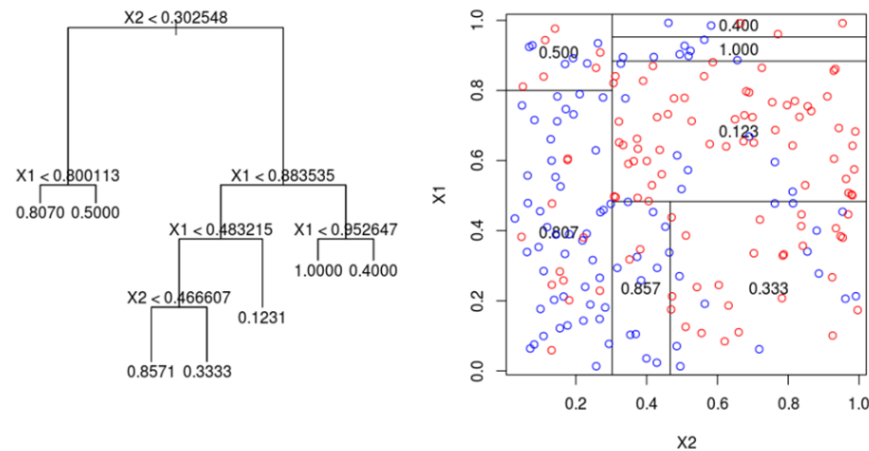
Thông số hiệu chỉnh mô hình: α

2.3.1.2. Decision Tree Regression

Giới thiệu mô hình

Cây quyết định (DT) là một phương pháp học có giám sát phi tham số được sử dụng để phân loại và hồi quy. Mục tiêu là tạo ra một mô hình dự đoán giá trị của biến mục

tiêu bằng cách học các quy tắc quyết định đơn giản được suy ra từ các tính năng dữ liệu.



Hình 2.7 Minh họa cây quyết định trong hồi quy

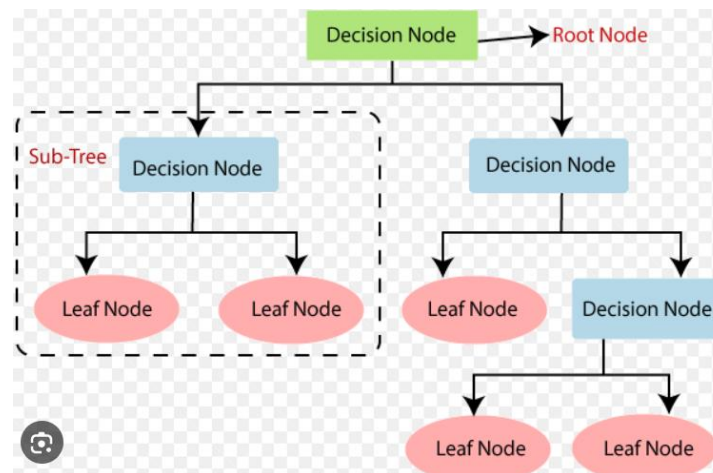
Một số ưu điểm của cây quyết định là:

- Đơn giản để hiểu và giải thích, dễ hình dung
- Có thể xác nhận một mô hình bằng cách sử dụng các bài kiểm tra thống kê. Điều đó làm cho nó có thể tính đến độ tin cậy của mô hình
- Chi phí sử dụng cây (nghĩa là dự đoán dữ liệu) là logarit theo số lượng điểm dữ liệu được sử dụng để huấn luyện cây (thời gian huấn luyện ngắn)

Nhược điểm của cây quyết định bao gồm:

- Cây quyết định có thể tạo ra các cây quá phức tạp không tổng quát hóa dữ liệu tốt. Điều này được gọi là trạng bị quá khớp so với dữ liệu huấn luyện. Các cơ chế như cắt tỉa, đặt số lượng mẫu tối thiểu cần thiết tại một nút lá hoặc đặt độ sâu tối đa của cây là cần thiết để tránh vấn đề này. Điều này sẽ được đề cập đến ở phần tham số mô hình
- Cây quyết định có thể không ổn định vì các biến thể nhỏ trong dữ liệu có thể dẫn đến việc tạo ra một cây hoàn toàn khác.
- Cây quyết định tạo cây thiên vị nếu một số lớp chiếm ưu thế. Do đó, nên cân bằng tập dữ liệu trước khi khớp với cây quyết định.

Các tham số đầu vào để cấu hình mô hình



Hình 2.8 Hình dạng chung của cây quyết định

Các tham số ta cần chú ý khi huấn luyện 1 cây quyết định:

- *max_depth*: Độ sâu cây quyết định
- *min_samples_split*: Số lượng mẫu tối thiểu cần thiết để tách một nút bên trong
- *min_samples_leaf*: Số lượng mẫu tối thiểu cần có tại một nút lá. Một điểm phân chia ở bất kỳ độ sâu nào sẽ chỉ được xem xét nếu nó để lại ít nhất *min_samples_leaf* mẫu huấn luyện trong mỗi nhánh trái và phải. Điều này có thể có tác dụng làm trơn mô hình, đặc biệt là trong hồi quy.

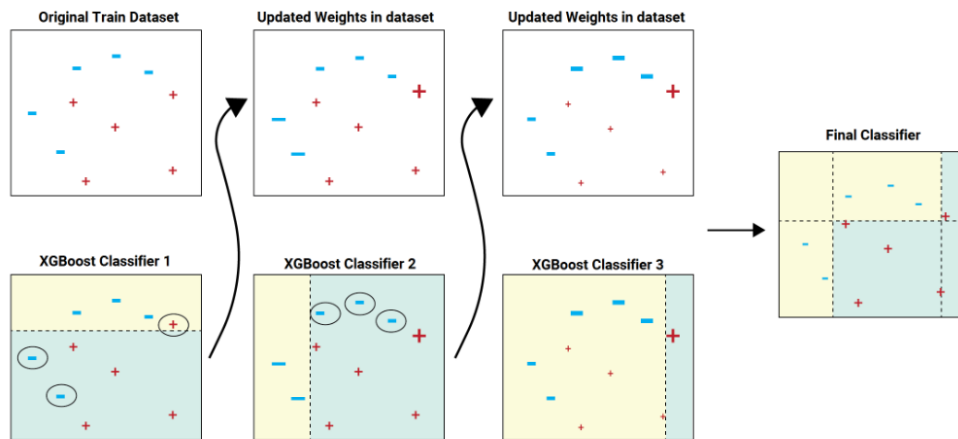
2.3.1.3. Xgboost Regression

Giới thiệu mô hình

Gradient Boosting đề cập đến một lớp thuật toán học máy tập hợp có thể được sử dụng để phân loại hoặc các vấn đề về mô hình dự đoán hồi quy.

Các tập hợp được xây dựng từ các mô hình cây quyết định. Các cây lần lượt được thêm vào quần thể và phù hợp để sửa các lỗi dự đoán của các mô hình trước đó. Đây là một loại mô hình máy học tập hợp được gọi là tăng cường.

Các mô hình phù hợp bằng cách sử dụng bất kỳ hàm mất mát khả vi tùy ý và thuật toán tối ưu hóa độ dốc gốc nào. Điều này đặt tên cho kỹ thuật này là “tăng cường độ dốc”, vì độ dốc mất mát được giảm thiểu khi mô hình phù hợp, giống như một mạng lưới thần kinh.



Hình 2.9 Minh họa Xgboost Regression

Các tham số đầu vào để cấu hình mô hình

- *n_estimators*: Số lượng cây trong quần thể, thường tăng lên cho đến khi không thấy sự cải thiện nào nữa.
- *max_depth*: Độ sâu tối đa của mỗi cây, giá trị thường nằm trong khoảng từ 1 đến 10.
- *eta*: Tỷ lệ học tập được sử dụng để tính trọng số cho từng mô hình, thường được đặt thành các giá trị nhỏ như 0,3, 0,1, 0,01 hoặc nhỏ hơn.
- *subsample*: Số lượng mẫu (hàng) được sử dụng trong mỗi cây, được đặt thành giá trị từ 0 đến 1, thường là 1,0 để sử dụng tất cả các mẫu.
- *colsample_bytree*: Số tính năng (cột) được sử dụng trong mỗi cây, được đặt thành giá trị từ 0 đến 1, thường là 1,0 để sử dụng tất cả các tính năng.

2.3.1.4. Nhận xét

Các mô hình thuần hồi quy tuyến tính cho ra kết quả tương tự nhau:

$$OLS Regression = Lasso Regression = Ridge Regression$$

Đối với các mô hình trên thì qua kết quả ta thấy được miền giá trị dự đoán điểm học phần có 1 phần nhỏ hơn 0 và 1 số dự đoán có giá trị lớn hơn 10 điều này là vô lí khi so sánh với kết quả thực tế (134 mẫu vi phạm nguyên tắc/36724 mẫu dữ liệu) điều này là 1 phần ngoại lệ khi sử dụng mô hình hồi quy tuyến tính. Phổ điểm thực tế của

bộ dữ liệu thực tế khoảng từ 5 đến 10, còn phổ điểm kết quả dự đoán chiếm phần lớn từ 6 đến 9. Kết quả thực tế dựa vào bảng so sánh RMSE (1 thước đo đánh giá trong mô hình hồi quy) thì cho ta kết quả tốt tầm khoảng [6,9] điểm ($RMSE < 1.5$), các mốc điểm từ [1,5] cho kết quả RMSE khá là cao (> 2) nhưng trên thực tế thì do chúng chiếm số lượng khá là nhỏ so với các khoảng còn lại nên cũng không ảnh hưởng nhiều đến kết quả chung mô hình lắm. Vậy khoảng dự đoán ổn áp nhất khi sử dụng cho mô hình này là điểm số sinh viên > 5 thì sẽ không có chênh lệch nhiều ($RMSE < 2$)

Đối với các mô hình sử dụng cây quyết định như Decision Tree Regressor hay Xgboost Regressor có kết quả tốt hơn chút do 1 phần là khoảng giá trị dự đoán của các mô hình này nhỏ hơn khoảng giá trị điểm thực tế do ảnh hưởng của thuật toán cây quyết định dùng để hồi quy. Còn các phổ điểm còn lại thì khá giống với các mô hình hồi quy trên, khoảng dự đoán tốt là từ [6,9], tốt là [5,10].

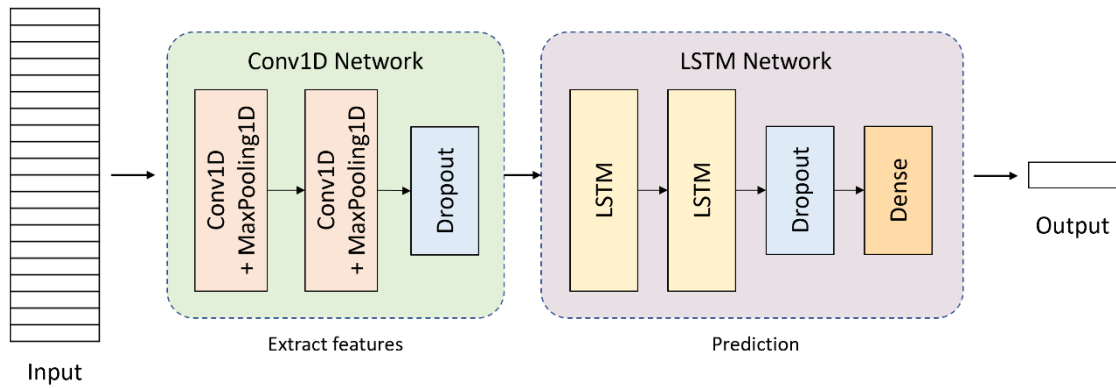
2.3.2. Hướng tiếp cận Neural Network

Tổng quan thuật toán

Thực hiện dự đoán kết quả học tập bằng mô hình kết hợp Conv1D (Convolutional 1D) và LSTM (Long Short-Term Memory). Mô hình được xây dựng trên ý tưởng kết hợp khả năng trích xuất đặc trưng không gian của Conv1D và khả năng mô hình hóa sự phụ thuộc vào thời gian của LSTM.

- **Input:** Đầu vào của mô hình là các đặc trưng bao gồm: thông tin về kết quả môn học của sinh viên (mã số sinh viên, mã môn học, học kỳ, năm học,...), thông tin về sinh viên (khoa, hệ đào tạo, điểm trung bình từng học kỳ,...) và thông tin về môn học (điểm trung bình môn học theo từng học kỳ, số lượng sinh viên đăng ký học thống kê theo trạng thái bình thường, trả nợ hay cải thiện,...)
- **Output:** Đầu ra là điểm môn học của sinh viên

Các lớp trong mô hình



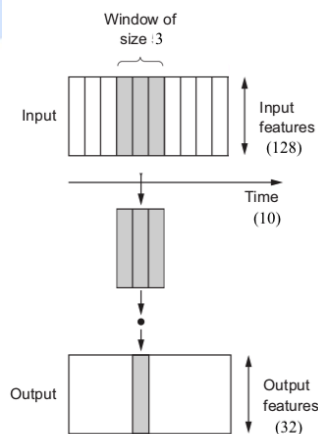
Hình 2.10 Minh họa cấu trúc mô hình Conv1D-LSTM

Cấu trúc của mô hình bao gồm các lớp Conv1D và LSTM, là hai loại lớp quan trọng trong lĩnh vực mạng nơ-ron học sâu.

a) Conv1D (Convolutional 1D) Layer

Cách thức hoạt động của lớp Conv1D là sử dụng bộ lọc để quét qua chuỗi dữ liệu, tìm kiếm các mẫu và đặc trưng quan trọng. Mỗi bộ lọc sẽ nhân tổ hợp tuyến tính của các giá trị trong một phần của chuỗi dữ liệu, từ đó tạo ra một giá trị mới đại diện cho mẫu hoặc đặc trưng tương ứng. Bằng cách di chuyển bộ lọc này, ta thu được một tập hợp các giá trị đại diện cho các mẫu không gian quan trọng.

Lớp Conv1D chủ yếu có vai trò trích xuất đặc trưng không gian từ dữ liệu. Các đặc trưng này có thể là những xu hướng, mẫu hoặc đặc điểm quan trọng liên quan đến quá trình học tập. Các đặc trưng này sẽ được chuyển tiếp đến các lớp sau đó để được sử dụng trong việc dự đoán kết quả học tập.

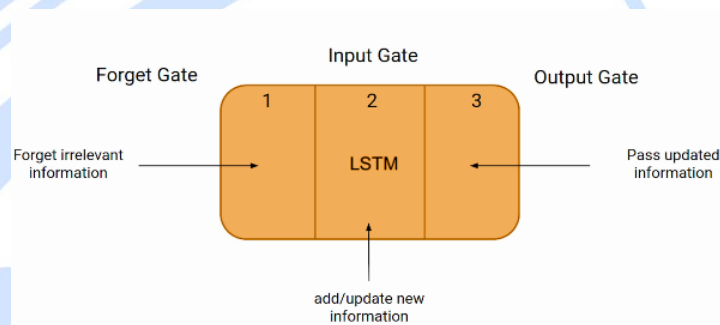


Hình 2.11 Minh họa phương thức hoạt động của Conv1D

b) *LSTM (Long Short-Term Memory) Layer*

Lớp LSTM có khả năng lưu trữ thông tin từ các bước thời gian trước và sử dụng nó để ảnh hưởng đến các bước thời gian sau. Điều này được thực hiện thông qua việc sử dụng các cổng (gate) để kiểm soát luồng thông tin.

- Cổng quên (forget gate) cho phép mô hình quyết định xem thông tin từ các thời điểm trước đó có nên được lưu trữ hay không.
- Cổng đầu vào (input gate) kiểm soát việc thêm thông tin mới vào trạng thái ẩn của LSTM dựa trên thông tin từ thời điểm hiện tại.
- Cổng đầu ra (output gate) quyết định phần nào của trạng thái ẩn sẽ được truyền ra ngoài để đưa ra dự đoán.



Hình 2.12 Minh họa phương thức hoạt động của LSTM

Khi áp dụng vào ngữ cảnh dự đoán điểm môn học, lớp LSTM có thể học được các mẫu và xu hướng trong dữ liệu điểm môn học của sinh viên theo thời gian, bao gồm sự tương quan giữa các điểm dữ liệu trong quá khứ và hiện tại.

c) *Các lớp khác (MaxPooling, Dense và Dropout)*

Các lớp này được sử dụng để kết hợp và biểu diễn đặc trưng đã học từ các lớp trước đó, học các biểu diễn phức tạp của dữ liệu, và giảm overfitting bằng cách chống lại sự phụ thuộc quá mức vào các đặc trưng cụ thể. Lớp MaxPooling1D được sử dụng để giảm kích thước của đầu ra từ lớp Conv1D bằng cách lấy giá trị lớn nhất trong mỗi cửa sổ dữ liệu. Điều này giúp giảm số lượng tham số và tăng tính tổng quát của mô hình.

Cách thức hoạt động

Quá trình huấn luyện mô hình:

- *Bước 1:* Dữ liệu được chuẩn bị và chia thành tập huấn luyện và tập kiểm tra.
- *Bước 2:* Mô hình được xây dựng với các lớp Conv1D, LSTM và các lớp khác như Dense và Dropout.
- *Bước 3:* Dùng RandomSearch để tìm kiếm siêu tham số tối ưu cho các tham số như *filters*, *cnn_kernel_size*, *lstm_units*, ...
- *Bước 4:* Mô hình được biên dịch với hàm loss là Mean Squared Error và tối ưu hóa bằng thuật toán Adam.
- *Bước 5:* Quá trình huấn luyện mô hình được thực hiện trên tập huấn luyện với số lượng *epoch* và kích thước *batch* được xác định trước.

Quá trình dự đoán và đánh giá:

- *Bước 1:* Mô hình được sử dụng để dự đoán điểm môn học trên tập kiểm tra.
- *Bước 2:* Kết quả dự đoán được so sánh với giá trị thực tế để đánh giá mô hình.
- *Bước 3:* Đánh giá kết quả dự đoán bằng Root Mean Squared Error (RMSE).

2.3.3. Hướng tiếp cận hệ khuyến nghị (Recommendation System)

2.3.3.1. Hệ khuyến nghị (Recommendation System)

Định nghĩa

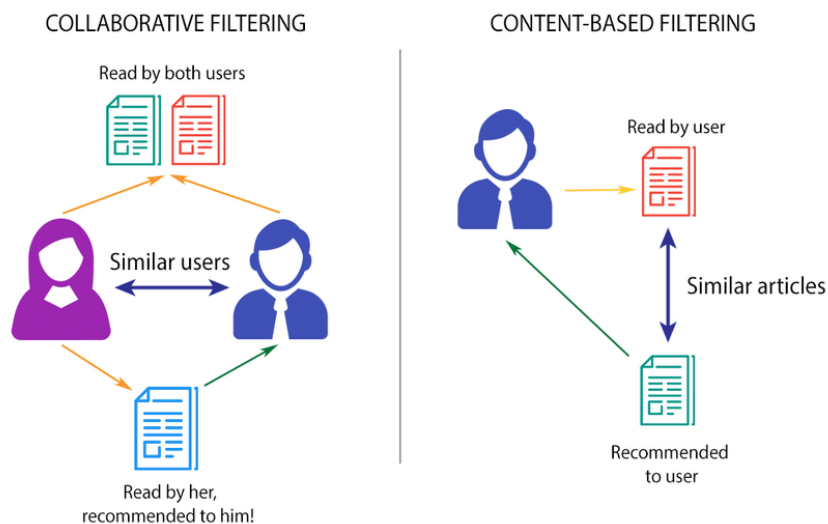
Recommendation System là một trong những ứng dụng phổ biến nhất của khoa học dữ liệu ngày nay. Chúng được sử dụng để dự đoán "rating" hoặc "preference" mà người dùng sẽ dành cho một mặt hàng. Hầu hết mọi công ty công nghệ lớn đều đã áp dụng chúng dưới một số hình thức.

Các mô hình hệ khuyến nghị

- *Simple Recommenders:* Đưa ra các đề xuất tổng quát cho mọi người dùng, dựa trên mức độ phổ biến và/hoặc thể loại phim. Ý tưởng cơ bản đằng sau hệ

thống này là những bộ phim nổi tiếng hơn và được giới phê bình đánh giá cao hơn sẽ có xác suất được khán giả bình thường thích cao hơn. Một ví dụ có thể là IMDB Top 250.

- Content-Based Recommenders
- Collaborative Filtering Recommenders
- Hybrid Recommenders: Hybrid Filtering là sự kết hợp của hai giải thuật Content-based Filtering và Collaborative Filtering: Hybrid Filtering được sử dụng mềm dẻo khi hệ thống Collaborative Filtering không có các hành vi (ratings), khi đó hệ thống sẽ sử dụng Content-Based Filtering và ngược lại, khi Content-Based Filtering không có các feature cần thiết trong việc đánh giá thì hệ thống sẽ sử dụng Collaborative Filtering để thay thế.



Phân tích các bước thực hiện

Trong bài toán này:

- Sinh viên giống như người dùng
- Môn học giống như sản phẩm
- Điểm môn học giống như đánh giá người dùng cho sản phẩm

Để xây dựng một hệ thống có thể tự động đề xuất các kết quả học tập cho sinh viên dựa trên kết quả học tập của các sinh viên khác:

- Bước đầu tiên là tìm kiếm các sinh viên hoặc môn học tương tự.
- Bước thứ hai là dự đoán các điểm của các môn học chưa được học bởi một người dùng.

Dựa vào định hướng của bài toán, ta có những vấn đề:

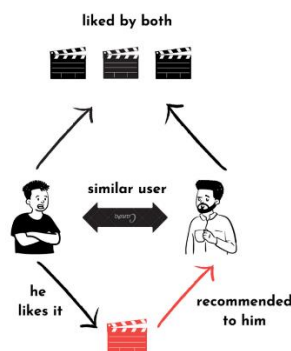
1. Làm thế nào để xác định những sinh viên hoặc môn học nào giống nhau?
2. Một khi ta đã biết những sinh viên nào giống nhau, làm thế nào để xác định điểm mà một sinh viên sẽ có được cho một môn học dựa trên các kết quả học tập của những sinh viên tương tự?

2.3.3.2. Collaborative Filtering

Định nghĩa

Collaborative Filtering là một kỹ thuật có thể dự đoán ra kết quả học tập những môn học mà sinh viên chưa học dựa trên kết quả học tập của các sinh viên tương tự.

Nó hoạt động bằng cách tìm kiếm một nhóm lớn sinh viên và tìm một tập con nhỏ hơn các sinh viên có kết quả học tập tương tự với một sinh viên cụ thể. Nó xem xét các môn học mà họ đã học và kết hợp chúng để tạo ra một danh sách gợi ý được xếp hạng.



Hình 2.13 Minh họa cách hoạt động của Collaborative Filtering

Collaborative Filtering là một họ thuật toán trong đó có nhiều cách để tìm kiếm sinh viên hoặc môn học tương tự và nhiều cách để tính toán kết quả học tập dựa trên các kết quả của những người dùng tương tự.

Ở đây, sự tương đồng không được tính bằng các yếu tố tính chất của sinh viên, và môn học. Nó được tính toán chỉ dựa trên đánh giá (rõ ràng hoặc ngụ ý) kết quả học tập của sinh viên dựa trên môn học. Ví dụ, hai sinh viên có thể được coi là tương tự nếu họ đưa ra cùng có kết quả học tập dựa trên cùng môn mặc dù có sự khác biệt lớn về thông tin cá nhân của 2 sinh viên đó.

Các hướng tiếp cận

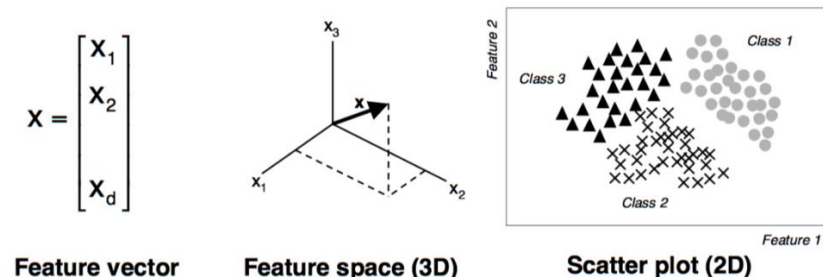
a) Memory based (Sử dụng toàn bộ bộ dữ liệu để dự đoán)

- **User-based:** Với mỗi sinh viên S, với một tập các sinh viên được cho là tương đồng với S dựa vào kết quả học tập ở các môn học trước. Ta cần dự đoán điểm cho môn học C (chưa được học qua), bằng cách chọn ra N sinh viên tương đồng nhất đã học môn đó và tính điểm dựa trên điểm của N sinh viên này
- **Item-based:** Với mỗi môn học C, với một tập các môn học được cho là tương đồng với C dựa vào kết quả học tập của các sinh viên đã học trước. Ta cần dự đoán điểm cho sinh viên S chưa học môn đó, bằng cách chọn ra N môn học tương đồng nhất với môn học đó mà sinh viên này đã học và dùng đó để dự đoán.

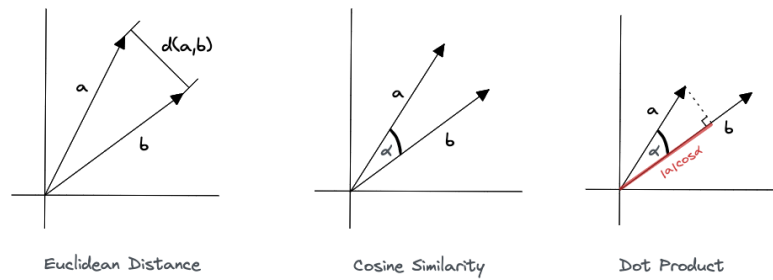
Cả hai phương pháp tuy hướng tiếp cận có đôi chút khác nhưng phương pháp thực hiện sẽ giống nhau nên nhóm đại biểu cách thực hiện User-based.

• Phương pháp tính độ tương đồng giữa các sinh viên

Mỗi sinh viên sẽ có một vector điểm các môn đã học. Và ta có thể biểu diễn vector này lên không gian nhiều chiều. Tới đây ta dùng các độ đo để gom cum các vector này.



Similarity Metrics



Hình 2.14 Ba thang đo tiêu biểu để tính độ tương đồng

- **Phương pháp dự đoán điểm dựa vào các sinh viên tương đồng**

Sau khi đã lấy được danh sách sinh viên tương đồng với sinh viên S. Ta chọn N sinh viên liên quan nhất. Sau đó với mỗi sinh viên, lấy điểm học tập P của môn học C và độ tương đồng U.

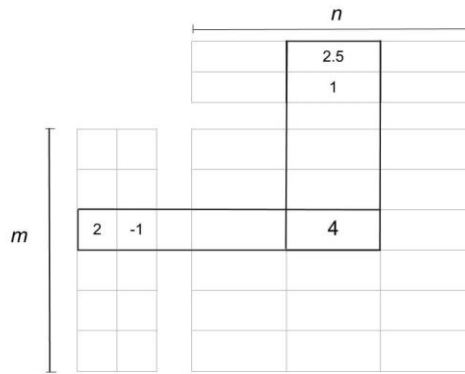
Khi đó, kết quả học tập môn học C của sinh viên S được dự đoán bằng:

$$P_C = \left(\sum_{c=1}^N P_c * U_c \right) / \left(\sum_{c=1}^N U_c \right)$$

b) Model based (Làm giảm hoặc nén bộ dữ liệu ma trận thưa giữa sinh viên - môn học)

Nếu mà ma trận kết quả học tập giữa môn học và sinh viên thưa thớt nhiều, việc làm làm giảm kích thước ma trận sẽ đẩy nhanh việc tính toán và đỡ bộ nhớ tính toán

Matrix factorization được xem như là quá trình phân tách ma trận lớn thành các ma trận nhỏ hơn. Ví dụ một ma trận A có kích thước $m \times n$ có thể được tác ra thành 2 ma trận X và Y với số chiều $m \times p$ và $p \times n$. Tùy thuộc vào thuật toán phân rã cụ thể mà số ma trận con được phân tách có thể nhiều hơn 2.



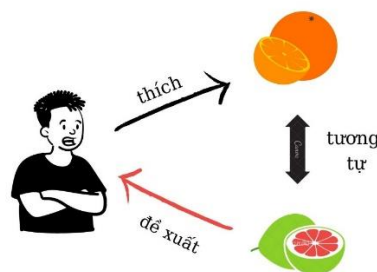
Hình 2.15 Minh họa Matrix Factorization

Các ma trận thành phần sẽ cho ta cái nhìn nhận sâu hơn về từng thành phần sinh viên và môn học. Ngoài ra ta có thể mở rộng thông tin ma trận bằng các thông tin đặc tính sinh viên và môn học, khi này số lượng thành phần ma trận tiềm ẩn sẽ tăng lên, giúp cải thiện kết quả dự đoán. Tuy nhiên cũng có thể dẫn tới overfitting.

Ta có thể thực hiện phân rã ma trận bằng thuật toán **SVD (Singular Value Decomposition)**.

2.3.3.3. Content Filtering

Đối với Content Filtering, hệ thống sẽ dựa vào các đặc tính của sinh viên (khoa, điểm trung bình qua các kỳ, số tín chỉ từng kỳ,...) và môn học (môn học khoa nào, số lượng người học qua môn, số lượng người học lại, điểm trung bình các sinh viên học,...) để từ đó đề xuất ra các sinh viên và môn học có đặc trưng gần tương tự.



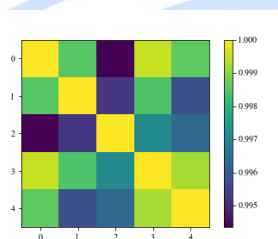
Hình 2.16 Một người thích cam, thì ta sẽ đề xuất người dùng các món có tính chua là bưởi

Phương pháp 1: Đề xuất điểm dựa trên điểm các môn học tương tự mà sinh viên đó đã học

Cách tính độ tương đồng giữa hai môn học

Do cố tình, trong quá trình chuẩn bị dữ liệu, các trường thuộc tính đã được Label Encoder hoặc One-hot Encoder. Nên bước đầu chỉ cần Minmax Scale cho từng cột dữ liệu là đã sẵn sàng cho việc tính tương đồng.

Ta sẽ tính *cosine similarity* cho từng cặp môn học trong đó và cho ra bảng ma trận độ tương đồng giữa các môn học.



Hình 2.17 Minh họa cho pairwise cosine similarity

Với ma trận tương quan, ta có thể thấy được 2 điều, thứ nhất độ mạnh tương quan thông qua màu sắc ta quy định, thứ hai ta có thể tìm và sắp xếp lại độ tương quan của môn học theo hàng của ma trận tương quan

Phương pháp 2: Đề xuất điểm dựa trên điểm các sinh viên tương tự đã học qua môn học đó

Các tính độ tương đồng giữa hai sinh viên: Tương tự như các tính độ tương đồng hai môn học

Phương pháp tính điểm dựa trên độ tương đồng

Tương tự phương pháp tính điểm dựa trên độ tương đồng phương pháp Collaborative Filtering – Memory based.

3. Cài đặt thực nghiệm

3.1. Dataset

Do cả ba phương pháp mà nhóm đề xuất đều được tổ chức khác nhau, và yêu cầu khác nhau về việc xây dựng dữ liệu sao cho phù hợp nhất với mô hình. Nên thực nghiệm của cả ba phương pháp là ở trên ba bộ dữ liệu khác nhau:

- *Recommendation System*: chạy trên ba bảng sinhvien (**17881** mẫu dữ liệu), monhoc (**639** mẫu dữ liệu) và diem (**593367** mẫu dữ liệu).
- *Machine Learning*: chạy trên ml_data (**632354** mẫu dữ liệu).
- *Neural Network*: chạy trên dl_data (**439320** mẫu dữ liệu).

3.2. Phương pháp đánh giá

Báo cáo này sử dụng độ đo Root Mean Squared Error (RMSE) để đánh giá hiệu suất của mô hình và độ chính xác của dự đoán.

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (\hat{G}_{ij} - G_{ij})^2}{N}}$$

RMSE tính toán sai số trung bình giữa giá trị dự đoán và giá trị thực tế bằng cách lấy căn bậc hai của tổng bình phương của các sai số. Điều này giúp đánh giá độ lớn của sai số mà mô hình dự đoán có trong việc ước lượng kết quả học tập.

Quá trình đánh giá bao gồm việc so sánh giá trị dự đoán của mô hình với giá trị thực tế trong tập dữ liệu kiểm tra. Giá trị RMSE càng gần với 0, tức là sai số càng nhỏ, thì mô hình càng có khả năng dự đoán chính xác điểm học tập.

3.3. Các phương pháp thực nghiệm

Toàn bộ các phương pháp đều được đánh giá dựa trên độ đo **Root Mean Squared Error (RMSE)**.

STT	Phương pháp	Đặc trưng chính
-----	-------------	-----------------

1	Linear Regression	Dự đoán điểm các môn học kỳ tiếp theo của sinh viên dựa trên mối quan hệ tuyến tính giữa các biến input và điểm học phần.
2	Ridge Regression	Sử dụng Linear Regression nhưng áp dụng một thành phần điều chuẩn (regularization) để giảm overfitting trong trường hợp dữ liệu có nhiễu hoặc đa cộng tuyến.
3	Lasso Regression	Tương tự như Ridge Regression, nhưng sử dụng một hình phạt L1 để thực hiện chọn lọc biến input quan trọng nhất cho dự đoán.
4	Decision Tree	Sử dụng cây quyết định để tạo ra các quy tắc dự đoán dựa trên các biến input. Các cây quyết định tách dữ liệu thành các nhánh con dựa trên các quy tắc tối ưu về độ tách (purity) của dữ liệu.
5	XgBoost	Một phương pháp kết hợp giữa cây quyết định và boosting, giúp tạo ra một mô hình mạnh mẽ và linh hoạt trong việc dự đoán điểm các môn học dựa trên các biến input.
6	Mô hình kết hợp Conv1D và LSTM	Xây dựng dựa trên ý tưởng kết hợp khả năng trích xuất đặc trưng không gian của Conv1D và khả năng mô hình hóa sự phụ thuộc vào thời gian của LSTM.
7	Collaborative Filtering – Model based	Dự đoán kết quả học tập những môn học mà sinh viên chưa học dựa trên kết quả học tập của các sinh viên tương tự bằng <i>phân rã ma trận</i> .
8	Collaborative Filtering – Memory based	Dự đoán kết quả học tập những môn học mà sinh viên chưa học dựa trên kết quả học tập của các sinh viên tương tự bằng <i>tính độ tương đồng</i> .
9	Content Filtering - Course based	Đề xuất điểm dựa trên điểm các môn học tương tự mà sinh viên đó đã học.
10	Content Filtering – Student based	Đề xuất điểm dựa trên điểm các sinh viên tương tự đã học qua môn học đó.

3.4. Kết quả thực nghiệm

STT	Phương pháp	RMSE Test Score
1	Collaborative Filtering – Model based	1.7200
2	Collaborative Filtering - Memory based	1.7258
3	Conv1D_LSTM	1.76854
4	Xgboost Regression	1.79939
5	Lasso Regression	1.88168
6	Ordinary Least Squares Regression	1.88256
7	Ridge Regression	1.88256
8	Content Filtering – Student based	1.8851
9	Decision Tree Regression	1.88814
10	Content Filtering - Course based	2.2207

4. Kết luận và hướng phát triển

Trong bài toán dự đoán điểm các môn học mà sinh viên lựa chọn cho học kỳ tiếp theo dựa trên dữ liệu lịch sử điểm cùng với thông tin về sinh viên và môn học, báo cáo này đã tiếp cận theo ba hướng: các mô hình máy học hồi quy, mô hình Neural Network (Conv1D và LSTM), và mô hình hệ thống khuyến nghị. Kết quả thực nghiệm cho thấy các phương pháp này đều có tiềm năng và tương đối hiệu quả trong việc dự đoán điểm môn học.

Các hướng phát triển tiếp theo của đề tài:

- Thử nghiệm thêm các phương pháp khác để cải thiện kết quả dự đoán

- Tối ưu hóa các siêu tham số của mô hình: tinh chỉnh các siêu tham số để đạt được hiệu suất tốt nhất bằng các phương pháp Grid Search, Random Search,...

5. Tài liệu tham khảo

- [1] H. R. J. L. A. J. Mack Sweeney, "Next-Term Student Performance Prediction," *Journal of Educational Data Mining*, vol. 8, 2016.
- [2] N. D. L. K.-G. A. a. S. L. Thai-Nghe, "Recommender System for Predicting Student," *In Proceedings of the 1st Workshop on Recommender*, vol. 1, 2010.
- [3] "Time-series Forecasting using Conv1D-LSTM : Multiple timesteps into future.," [Online]. Available: <https://shivapriya-katta.medium.com/time-series-forecasting-using-conv1d-lstm-multiple-timesteps-into-future-acc684dcaaa>.
- [4] "Lasso & Ridge Regression | A Comprehensive Guide in Python & R," [Online]. Available: <https://www.analyticsvidhya.com/blog/2017/06/a-comprehensive-guide-for-linear-ridge-and-lasso-regression/>.
- [5] "Comparing Decision Tree Algorithms: Random Forest vs. XGBoost," [Online]. Available: <https://www.activestate.com/blog/comparing-decision-tree-algorithms-random-forest-vs-xgboost/>.

6. Bảng phân công công việc

Nội dung	Thành viên	Bùi Nguyễn Anh Trung 20520332 (Nhóm trưởng)	Hồ Thanh Tĩnh 20520813	Nguyễn Trần Minh Anh 20520394	Lê Nguyễn Bảo Hân 20520174	Nguyễn Văn Đức Ngọc 20521666
Phân công, quản lý công việc chung		✓		✓		
Tiền xử lý dữ liệu		✓		✓		✓
Tìm hiểu phương pháp Recommendation System		✓				
Tìm hiểu phương pháp Machine Learning						✓
Tìm hiểu phương pháp Neural Network			✓	✓	✓	
Tổng hợp nội dung và định dạng báo cáo					✓	
Tổng hợp demo			✓			
Làm slide		✓	✓	✓	✓	✓
Làm video			✓			

Viết báo cáo phần phương pháp Recommendation System	✓				
Code, chạy thực nghiệm Recommendation System	✓				
Viết báo cáo phần phương pháp Neural Network				✓	
Code, chạy thực nghiệm Neural Network			✓	✓	
Viết báo cáo phần phương pháp Machine Learning					✓
Code, chạy thực nghiệm Machine Learning			✓		✓
Viết báo cáo xử lý dữ liệu			✓		
Định hướng đề án, góp ý, chỉnh sửa nội dung từng phần	✓				
Mức độ hoàn thiện (%)	100%	100%	100%	100%	100%