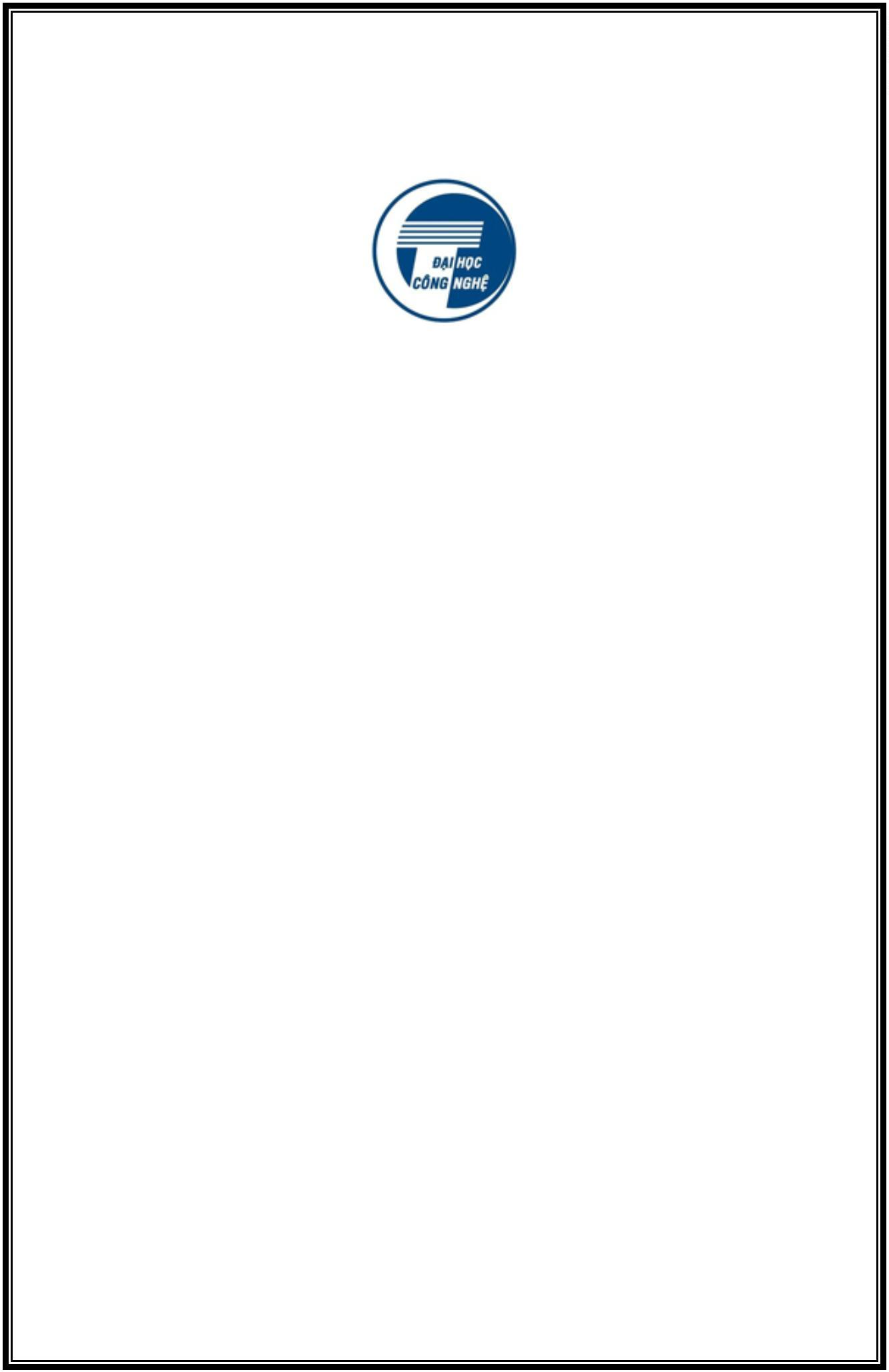
**ĐẠI HỌC QUỐC GIA HÀ NỘI**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

**KHOA: CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO MÔN: KHAI PHÁ DỮ LIỆU**

**DỰ ĐOÁN TUỔI NGƯỜI DÙNG THÔNG QUA CÁC NHÓM MÀ NGƯỜI DÙNG**

**ĐÃ THAM GIA**

|  |  |
| --- | --- |
| **Sinh viên thực hiện** | **Nguyễn Hữu Cứ - 22026532**  **Võ Quang Sáng - 22026526**  **Nguyễn Hoàng Điệp - 22026534**  **Lê Công Hoàng - 22026555**  **Trương Đức Quang - 22026536** |
| **Giảng Viên:** | **ThS. Vương Thị Hải Yến** |
| **MLHP:** | **INT3209 5** |

**Hà Nội - 2024**

[**PHẦN 1: ĐẶT VẤN ĐỀ VÀ XÁC ĐỊNH BÀI TOÁN**](#_5x1wn0qmx1uw) **5**

[1.1. Đặt vấn đề](#_l4s0jzdtoocl) 6

[1.2. Xác định bài toán](#_gc9jd2566d2) 6

[1.2.1. Phân tích yêu cầu](#_wxjj60a9vfps) 6

[1.2.2. Tập dữ liệu (Dataset)](#_8o9oo9pqwa2y) 7

[Hình 1.2.2.1. Tập dữ liệu train](#_ka5b59c5s9t1) 8

[Hình 1.2.1.2. Tập dữ liệu test](#_c43cu87fce09) 8

[1.2.3. Đề xuất mô hình:](#_2d85qb9t2qt4) 8

[1.2.4. Các tiêu chí đánh giá:](#_s35bzx5dephu) 10

[Hình 1 Hàm tính độ chấp nhận được](#_jmwp3mw7vjyl) 11

[**PHẦN 2: HIỂU DỮ LIỆU**](#_msxdn9xl2tvx) **12**

[**2.1. Tìm hiểu và phân tích dữ liệu**](#_nv7hzr771nd5) **12**

[2.1.1. Nhãn (Các lớp tuổi)](#_fb7o9jbdtoob) 12

[2.1.1.1. Thuộc tính:](#_thz8daet4sjc) 12

[2.1.1.2. Dạng dữ liệu:](#_ln6tfwly79co) 12

[2.1.1.3. Thống kê dữ liệu:](#_ffspdhjoh51x) 12

[Hình 2.1. Biểu đồ tần suất (a) và tỉ lệ phần trăm tương ứng (b) của nhãn dữ liệu trên tập train](#_dr8dk9xqhtt) 13

[Hình 2.2. Biểu đồ tần suất (a) và biểu đồ phần trăm tương ứng (b) của nhãn dữ liệu trên tập test](#_t8q3uy2c8gl5) 14

[2.1.2. Dữ liệu đặc trưng (Group\_id)](#_6edd9371cqrf) 14

[2.1.2.1.Trên toàn bộ dữ liệu](#_3l6zfq4m4ry8) 14

[2.1.2.1.1. Thuộc tính](#_ktdus7a06dbp) 14

[2.1.2.1.2. Dạng dữ liệu](#_7viwdsdzsna8) 15

[2.1.2.1.3. Thống kê dữ liệu](#_viuyb7c6l77g) 15

[Bảng 2.1. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất](#_239zglp1vlnk) 15

[Hình 2.3. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất](#_gbcp8ndr38td) 15

[Bảng 2.2. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10](#_dl5rkpa5ikwr) 16

[Hình 2.4. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id](#_35fkzq3w9iln) 17

[2.1.2.2. Trên các lớp dữ liệu.](#_h9153drev3nn) 18

[2.1.2.2.1. Lớp ‘18-24’](#_xcr3byvmudvs) 18

[Bảng 2.3. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất lớp ‘18-24’](#_tj6v9rfbp49p) 18

[Hình 2.5. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất lớp ‘18-24’](#_b9psv3m2l83q) 19

[Bảng 2.4. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10](#_74c8487fgg0e) 19

[Hình 2.6. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id lớp ‘18-24’](#_6kdm7sdy3foj) 20

[2.1.2.2.2. Lớp ’25-34’](#_lbwkvcl7pmms) 21

[Bảng 2.5. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất lớp ‘25-34’](#_pad5kezih44x) 21

[Hình 2.7. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất lớp ‘25-34’](#_xhrbu2o7c5on) 22

[Bảng 2.6. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10](#_qcq6byvquspp) 22

[Hình 2.8. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id lớp ‘25-34’](#_szgn3yffjkbf) 23

[2.1.2.2.3. Lớp ‘35-44’](#_tjlhbkc1zodn) 24

[Bảng 2.7. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất lớp ‘35-44’](#_oe7ffctisc81) 24

[Hình 2.9. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất lớp ‘35-44’](#_jugih9r4qycr) 24

[Bảng 2.8. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10](#_cq3f3d8350xn) 25

[Hình 2.10. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id lớp ‘35-44’](#_9mpf9eyxxy68) 25

[2.1.2.2.3. Lớp ‘45-54’](#_6ocvbr2xc8qq) 26

[Bảng 2.9. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất lớp ‘45-54’](#_bzhbbtt01ni9) 26

[Hình 2.11. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất lớp ‘45-54’](#_gwy326gnjdai) 27

[Bảng 2.10. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10](#_qnginldior5z) 27

[Hình 2.12. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id lớp ‘45-54’](#_d8nzctmij0e5) 28

[2.1.2.2.4. Lớp ‘55+’](#_58kgthnwqjsd) 29

[Bảng 2.11. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất lớp ‘55+’](#_eewl1lbfi9xp) 29

[Hình 2.13. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất lớp ‘55+’](#_3jtm5ls4e1ex) 29

[Bảng 2.12. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10](#_68nvngxv9isp) 30

[Hình 2.14. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id lớp ‘55+’](#_ragaasf8mrbo) 30

[**PHẦN 3: TIỀN XỬ LÝ DỮ LIỆU VÀ BIỂU DIỄN DỮ LIỆU**](#_kh5870sk5ylx) **32**

[3.1. Trên toàn bộ dữ liệu](#_131l5frph7ot) 32

[3.1.1. Làm sạch dữ liệu](#_4ew4siv9agee) 32

[3.1.1.1. Đánh giá các yếu tố](#_75xktduse4ub) 32

[3.1.1.2. Cách xử lý](#_y3slrxqh7cyq) 32

[Hình 3.1. Đoạn mã sử dụng phương pháp IQR](#_lwxtvrl8dcep) 33

[Hình 3.2. Đoạn mã sử dụng phương pháp dùng mô hình để loại bỏ điểm ngoại lai](#_42wq88ko93o8) 34

[Vấn đề 1: Loại bỏ các giá trị nhiễu bằng phương pháp IQR](#_fcszl2979btw) 34

[Vấn đề 2: Sử dụng mô hình logistic Regression để dự đoán ra các điểm ngoại lại và bỏ các giá trị đó](#_6qphniold15e) 34

[3.1.2. Cân bằng và tích hợp dữ liệu](#_7v542rhw4am8) 34

[Hình 3.3. Hàm sinh mẫu ngẫu nhiên dựa trên xác suất](#_vsoxuslq9ie) 35

[3.1.3. Biểu diễn dữ liệu.](#_otw17zyzd1sn) 35

[3.1.3.1. Nhãn (Label):](#_ydr7e8vlovw5) 35

[Hình 3.4. Các giá trị đại diện của label](#_wvz3its8lh60) 36

[3.1.3.2. Các đặc trưng từ group\_id](#_k1aifuljkobx) 36

[3.1.3.2.1. Sử dụng thư viện có sẵn để trích xuất ra đặc trưng](#_i8ujqg6svfv0) 36

[Hình 3.5.Đoạn mã xử lý các group\_id thành vector đặc trưng](#_soh19xxk16xt) 36

[3.1.3.2.2. Lựa chọn danh sách group\_id làm dữ liệu phân loại nhị phân](#_zh0c0yu65avq) 36

[Hình 3.6. Đoạn mã xử lý đặc trưng](#_t1a10pyi00ap) 37

[Hình 3.7. Minh họa ma trận sau khi đã biểu diễn với cột đầu tiên là nhãn](#_eaw3z7edl8t1) 37

[3.1.3.2.3. Các đặc trưng mới](#_961o3n7k7k27) 37

[Vấn đề 3: Tạo dữ liệu đặc trưng từ số lượng nhóm tham gia của 1 người dùng.](#_73e0l29z14ee) 37

[Vấn đề 4: Tạo dữ liệu đặc trưng từ các nhóm trùng nhau giữa các lớp.](#_dmuy41tl6fr4) 38

[Vấn đề 5:Tạo dữ liệu đặc trưng từ độ dài của group\_id](#_enhzp4shqqle) 39

[Vấn đề 6: Tạo ra các luật (rulesets) để dùng làm đặc trưng dự đoán cho mô hình sử dụng CountVectorizer làm đặc trưng](#_wkn8xywcsclj) 39

[3.2. Trên các lớp dữ liệu](#_x5ue4qiyj5d9) 39

[3.2.1. Biểu diễn dữ liệu](#_ihmgku9o3f27) 40

[3.2.1.1. Các đặc trưng từ group\_id](#_ceoi33vvj6pg) 40

[Hình 3.12. Đoạn mã lấy n phần tử xuất hiện nhiều nhất ở trong một lớp](#_y46y4gpdtqji) 40

[**PHẦN 4: ĐÁNH GIÁ CÁC MÔ HÌNH**](#_skgloyuh3hab) **41**

[Hình 4.1. Kiến trúc mô hình giải quyết bài toán](#_qxoda3opwim1) 41

[4.1. Mô hình Logistic Regression](#_tfn8tbys45j7) 41

[4.1.1. Khi sử dụng đặc trưng trích xuất từ thư viện](#_x5m8km5ohupw) 41

[Hình 4.2. Đoạn mã sử dụng classWeight và k-fold cho mô hình](#_8bzy7oaq1u4k) 42

[Hình 4.3. Kết quả của mô hình khi sử dụng cross validation với k = 5 và class weight trên tập train và tập test](#_m5jm0ekbpktx) 43

[Hình 4.4. Ma trận nhầm lẫn tương ứng](#_jum671tibu2c) 44

[4.1.2. Khi sử dụng đặc trưng theo cách tiếp cận lấy từ hàm matrix feature](#_w7j09hed4v6h) 44

[4.1.2.1. Chưa đề cập đến các vấn đề](#_hlm95nkcqmgc) 44

[Bảng 4.1. So sánh giữa 2 cách lấy số nhóm làm đặc trưng với số lượng nhóm trung bình](#_xuqojxlti0dc) 45

[Bảng 4.2. So sánh hiệu quả mô hình trên tập train và tập test](#_o5u48nwvbgti) 46

[4.1.2.2. Cân bằng mẫu](#_joi7tsbluwl3) 47

[4.1.2.2.1. Sử dụng hàm](#_z3aqlafeqkuh) 47

[Bảng 4.3. Kết quả mô hình khi sử dụng hàm để tăng mẫu tối thiểu](#_71a9iqj6h50d) 48

[4.1.2.2.2. Sử dụng thư viện SMOTE](#_670f0cmsi7e0) 49

[Hình 4.5. Kết quả của mô hình khi sử thư viện smote để cân bằng mẫu](#_ss0pebgwxk36) 49

[4.1.2.3. Vấn đề 1 và Vấn đề 3.](#_sv72a7pa0u1a) 49

[Hình 4.6. Kết quả của mô hình sau loại bỏ giá trị ngoại lai](#_7k7vg252k0uj) 50

[Hình 4.7. Kết quả của mô hình khi sử dụng số lượng nhóm tham gia của người dùng](#_21un8hwv8g8s) 51

[4.1.2.4. Vấn đề 2](#_5ledqnmty7ly) 51

[Hinh 4.8. Kết quả của mô hình khi sử dụng mô hình Logistic để loại bỏ điểm ngoại lệ](#_hkdvykdjvc6q) 51

[Hình 4.9. Kết quả của mô hình khi kiểm tra lại trên tập train](#_gd23lkm7j37) 52

[4.1.2.6. Vấn đề 4](#_y19j36uqu3yl) 52

[Hình 4.10. Kết quả của mô hình khi loại bỏ đi phần tử trùng nhau giữa 2 lớp kề nhau với số lượng nhóm tốt nhất là 15.](#_mlugyh370m7m) 52

[4.1.2.7. Vấn đề 5](#_i3e3mz20u1d8) 53

[Hình 4.11. Kết quả của mô hình sau khi thêm đặc trưng từ độ dài id của nhóm](#_al6faxhdzf6z) 53

[4.2. Mô hình SVM (Support Vector Machine)](#_6r2zolgdrz7q) 54

[4.2.1. Khi sử dụng đặc trưng trích xuất từ thư viện](#_e411mbv90b8j) 54

[Hình 4.12. Đoạn mã sử dụng classWeight và k-fold cho mô hình](#_leqoro8iygye) 54

[Hình 4.13. Kết quả của mô hình khi sử dụng cross validation với k = 5 và class weight trên tập train và tập test](#_ynjw3vcr1mj) 55

[Hình 4.14. Ma trận nhầm lẫn tương ứng](#_moilsre6nmlk) 55

[4.3. Mô hình Random Forest](#_px40jri0bz06) 56

[Tập train Tập test](#_5vum2ikih2kk) 56

[4.4. Mô hình LightGBM:](#_qyk3pef386fo) 56

[Hình 4.15. Kết quả khi sử dụng mô hình Light gbm](#_355ncsghgsry) 57

[4.5.Mô hình Naïve Bayes](#_149z19i7n523) 57

[4.5.1. Khi sử dụng đặc trưng trích xuất từ thư viện](#_7soixc2iqnzq) 57

[Hình 4.16. Sử dụng RandomOverSampler và RandomUnderSampler cho tập dữ liệu](#_5fbx7v3ilhf7) 58

[Hình 4.17. Tập dữ liệu trước và sau khi chỉnh sửa](#_5sp0khowx3e5) 58

[Hình 4.18. Đoạn mã sử dụng tập dữ liệu mới và k-fold cho mô hình](#_6exb1dj7iehz) 58

[Hình 4.19. Kết quả của mô hình khi sử dụng cross validation với k = 5](#_6966dq4oh431) 59

[Hình 4.20. Ma trận nhầm lẫn tương ứng](#_5fr3mxvrarus) 59

[**PHẦN 5: TỔNG KẾT**](#_aw6s8vmopqrr) **60**

[5.1. Mô hình tốt nhất](#_uxwbvzrzcfr6) 60

[5.2. Các vấn đề còn tồn đọng](#_xwtvhheegdhw) 60

[5.3. Hướng cải thiện](#_cb4498en6f6h) 60

[**PHẦN 6: PHÂN CHIA CÔNG VIỆC**](#_gmqq4dkqx7lf) **61**

# 

# **PHẦN 1: ĐẶT VẤN ĐỀ VÀ XÁC ĐỊNH BÀI TOÁN**

## **1.1. Đặt vấn đề**

Trong thời đại số hóa ngày nay, việc dự đoán thông tin cá nhân của người dùng trở thành một phần quan trọng trong việc nâng cao trải nghiệm người dùng và tối ưu hóa các dịch vụ trực tuyến. Một trong những thông tin quan trọng nhất là tuổi của người dùng. Tuổi không chỉ ảnh hưởng đến sở thích và hành vi tiêu dùng mà còn là yếu tố quyết định trong việc tiếp cận thông tin và sản phẩm phù hợp. Tuy nhiên, việc thu thập và xác minh tuổi của người dùng thường gặp nhiều khó khăn do các vấn đề về quyền riêng tư và bảo mật. Vì vậy, dự đoán tuổi người dùng dựa trên các dữ liệu có sẵn trở thành một giải pháp tiềm năng và có thể mang lại nhiều hiệu quả.

Trong nghiên cứu này, chúng tôi sẽ tập trung vào việc dự đoán tuổi người dùng thông qua các nhóm mà họ đã tham gia. Các nhóm này được xác định thông qua dữ liệu *group\_id*, và tuổi của nhóm được biểu thị qua nhãn *age\_group*. Giả định rằng, mỗi nhóm có một đặc trưng riêng về độ tuổi của thành viên, chúng tôi sẽ sử dụng các mô hình học máy để phân tích và dự đoán tuổi của người dùng dựa trên sự tham gia của họ trong các nhóm này. Mục tiêu của nghiên cứu là xây dựng một mô hình dự đoán chính xác, góp phần cải thiện các dịch vụ cá nhân hóa và đáp ứng nhu cầu người dùng một cách hiệu quả hơn.

Với tập dữ liệu bao gồm các nhãn tuổi (***label***) và các mã định danh nhóm (***group\_id***), chúng tôi sẽ áp dụng các phương pháp học máy để phát hiện mối quan hệ giữa nhóm tham gia và độ tuổi của người dùng. Việc phân tích dữ liệu này sẽ giúp chúng tôi xác định các mẫu hình hành vi và mối liên hệ giữa nhóm và tuổi, từ đó tạo ra một mô hình dự đoán tuổi người dùng hiệu quả.

## **1.2. Xác định bài toán**

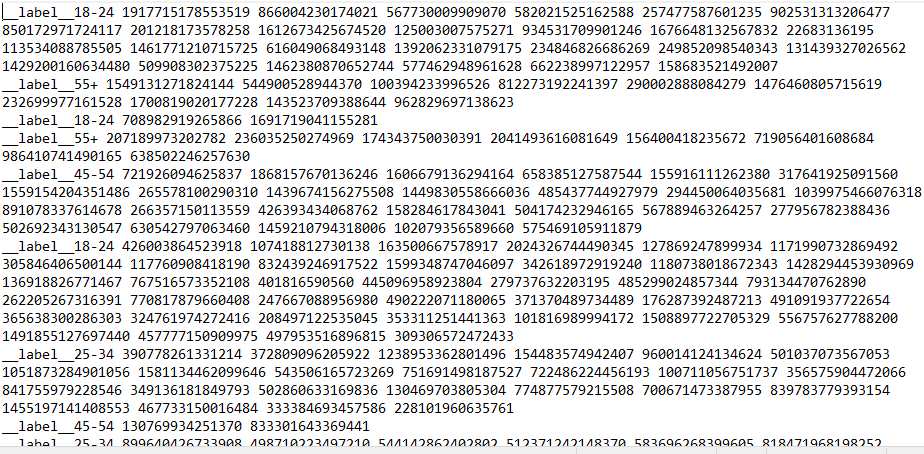
### **1.2.1. Phân tích yêu cầu**

Bài toán này yêu cầu chúng ta xác định nhóm tuổi của người dùng dựa trên các **group\_id** (nhóm) mà họ đã tham gia trên một mạng xã hội. Có thể thấy đây là một bài toán **phân loại (Classification)**, cụ thể là phân loại đa lớp, vì chúng ta cần gán mỗi người dùng dựa trên các nhóm mà người dùng đã tham gia vào một trong các nhóm tuổi xác định trước, bao gồm 5 nhóm tuổi: **Nhóm 1** - **‘18-24’**, **Nhóm 2** - **‘25-34’**, **Nhóm 3** - **‘35-44’**, **Nhóm 4** - **‘45-54’**, **Nhóm 5** - **‘55+’** . Các yêu cầu cụ thể của bài toán bao gồm:

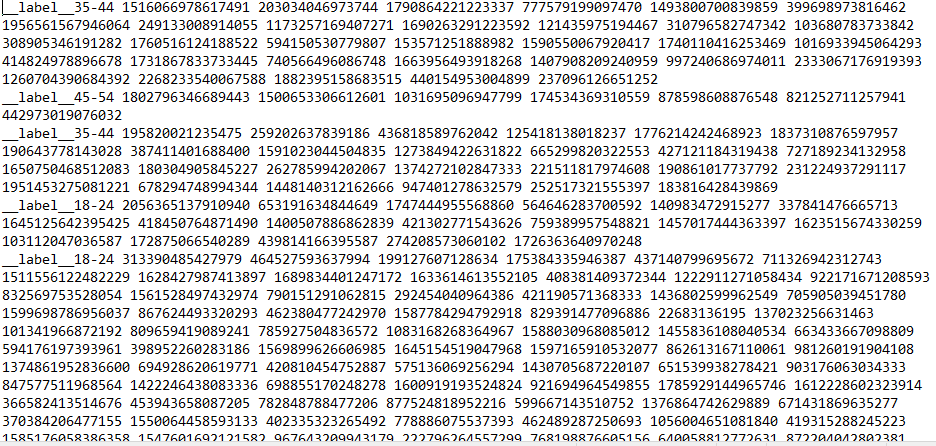
* **Phân loại**: Chia người dùng thành các nhóm tuổi dựa trên các đặc điểm về tham gia group.
* **Dự đoán nhóm tuổi**: Xác định nhóm tuổi của người dùng dựa trên dữ liệu về các group mà họ đã tham gia.

### **1.2.2. Tập dữ liệu (Dataset)**

Tập dữ liệu được sử dụng ở trong bài toán này bao gồm 2 phần chính : **Dữ liệu huấn luyện (Tập train)** và **dữ liệu kiểm tra (Tập test).**



#### *Hình 1.2.2.1. Tập dữ liệu train*



#### *Hình 1.2.1.2. Tập dữ liệu test*

* Cụ thể: Cả tập dữ liệu train và tập dữ liệu test bao gồm:
* Bao gồm các nhãn tuổi và danh sách các group\_id mà người dùng đã tham gia
* Các nhãn tuổi bao gồm:

1. **\_\_label\_\_18-24**: Người dùng từ 18 đến 24 tuổi.
2. **\_\_label\_\_25-34**: Người dùng từ 25 đến 34 tuổi.
3. **\_\_label\_\_35-44**: Người dùng từ 35 đến 44 tuổi.
4. **\_\_label\_\_45-54**: Người dùng từ 45 đến 54 tuổi.
5. **\_\_label\_\_55+** : Người dùng từ 55 tuổi trở lên.

* Danh sách các **group\_id** tương ứng với các nhãn: 1917715178553519,12242839239,.v.v,...

### **1.2.3. Đề xuất mô hình:**

Dựa trên bài toán phân lớp đa lớp chúng ta có thể đề xuất một số mô hình có thể sử dụng cho bài toán này:

1. Mô hình Logistic Regression:

* **Giới thiệu**: Logistic Regression là một mô hình thống kê dùng để dự đoán xác suất của một biến phụ thuộc nhị phân. Trong trường hợp phân lớp đa lớp, Logistic Regression có thể được mở rộng bằng cách sử dụng phương pháp One-vs-Rest (OvR) hoặc Softmax Regression.
* **Ưu điểm**: Đơn giản, dễ hiểu và dễ triển khai. Hiệu quả với các bài toán có số lượng đặc trưng không quá lớn.
* **Nhược điểm**: Hiệu suất có thể giảm khi dữ liệu có nhiều đặc trưng hoặc không tuyến tính.

1. Mô hình SVM (Support Vector Machine):

* **Giới thiệu**: SVM là một mô hình học máy mạnh mẽ dùng để phân loại dữ liệu bằng cách tìm ra một siêu phẳng tối ưu phân chia các lớp. Đối với phân lớp đa lớp, SVM có thể được mở rộng bằng cách sử dụng phương pháp One-vs-One hoặc One-vs-Rest.
* **Ưu điểm**: Hiệu quả với các bài toán có số lượng đặc trưng lớn và dữ liệu không tuyến tính (khi sử dụng kernel trick).
* **Nhược điểm**: Thời gian huấn luyện có thể lâu với các tập dữ liệu lớn. Cần chọn kernel và tham số phù hợp.

1. Mô hình Random Forest:

* **Giới thiệu**: Random Forest là một mô hình học máy dựa trên tập hợp các cây quyết định. Mỗi cây trong rừng được huấn luyện trên một tập con ngẫu nhiên của dữ liệu và kết quả cuối cùng được lấy trung bình từ các cây.
* **Ưu điểm**: Hiệu suất cao, ít bị overfitting, có khả năng xử lý dữ liệu thiếu và không cần chuẩn hóa dữ liệu.
* **Nhược điểm**: Khó giải thích, thời gian huấn luyện và dự đoán có thể lâu với các tập dữ liệu lớn

1. Mô hình LightGBM:

* **Giới thiệu**: LightGBM (Light Gradient Boosting Machine) là một mô hình học máy dựa trên phương pháp boosting, được thiết kế để tăng tốc độ và hiệu suất của quá trình huấn luyện.
* **Ưu điểm**: Hiệu suất cao, tốc độ huấn luyện nhanh, khả năng xử lý dữ liệu lớn và hỗ trợ nhiều loại dữ liệu khác nhau.
* **Nhược điểm**: Cần điều chỉnh nhiều tham số, có thể bị overfitting nếu không điều chỉnh đúng cách.

1. Mô hình Naïve Bayes

* **Giới thiệu**: Naïve Bayes là một mô hình phân loại dựa trên định lý Bayes với giả định rằng các đặc trưng là độc lập điều kiện. Mô hình này phù hợp với các bài toán phân lớp khi các đặc trưng không có mối quan hệ phức tạp.
* **Ưu điểm**: Đơn giản, nhanh chóng và hiệu quả với các tập dữ liệu lớn. Mô hình có khả năng hoạt động tốt với các đặc trưng độc lập và thường yêu cầu ít bộ nhớ.
* **Nhược điểm:** Giả định độc lập giữa các đặc trưng không luôn đúng trong thực tế, có thể dẫn đến hiệu suất kém khi có sự phụ thuộc giữa các đặc trưng. Mô hình cũng có thể gặp khó khăn với các tập dữ liệu có ít mẫu.

### **1.2.4. Các tiêu chí đánh giá:**

1. **Confusion Matrix (Ma trận nhầm lẫn):**

* **Giới thiệu:** Ma trận nhầm lẫn là một công cụ trực quan giúp đánh giá hiệu suất của mô hình phân loại. Nó hiển thị số lượng dự đoán đúng và sai cho từng lớp**.**
* **Ý nghĩa:** Giúp xác định các lỗi phân loại cụ thể và hiểu rõ hơn về cách mô hình hoạt động trên từng lớp.

Ở trong bài toán này, ta được phép chấp nhận các lớp phân loại sai không quá một lớp dựa trên ma trận nhầm lẫn. Ví dụ: Nếu như kết quả thực tế là nhóm 2 nhưng mô hình lại đoán là nhóm 1 hoặc nhóm 3 thì ta coi là chấp nhận được với độ chấp nhận được tính theo hàm sau:

|  |
| --- |
|  |
| Hình 1 Hàm tính độ chấp nhận được |

1. **Độ đo Macro:** với các chỉ số các chỉ số như: **Độ chính xác** (**Precision**), **Độ nhạy** (**Recall**), và **F1-Score**.

* **Giới thiệu**: Độ đo Macro tính toán các chỉ số như Độ chính xác (Precision), Độ nhạy (Recall), và F1-Score cho từng lớp riêng lẻ, sau đó lấy trung bình cộng của các chỉ số này.
* Các chỉ số:
* **Độ chính xác** **(Precision**): Tỷ lệ giữa số lượng dự đoán đúng trên tổng số mô hình dự đoán là đúng.
* **Độ nhạy** (**Recall**): Tỷ lệ giữa số lượng dự đoán đúng trên tổng số lượng đúng thực sự.
* **F1-Score**: Trung bình điều hòa của Precision và Recall, cung cấp một thước đo cân bằng giữa Precision và Recall.

# **PHẦN 2: HIỂU DỮ LIỆU**

# **2.1. Tìm hiểu và phân tích dữ liệu**

Với dữ liệu như trên và yêu cầu của bài toán là phân loại (phân lớp) đa lớp ta sẽ tiếp cận dữ liệu theo 2 phương diện là: **Nhãn** (Label) và **group\_id.**

### **2.1.1. Nhãn (Các lớp tuổi)**

#### **2.1.1.1. Thuộc tính**:

Là nhóm tuổi của người dùng, là loại thuộc tính phân loại có thứ tự (Ordinal Categorical Data)

#### **2.1.1.2. Dạng dữ liệu:**

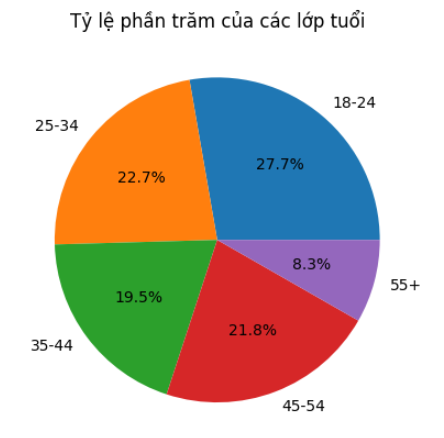
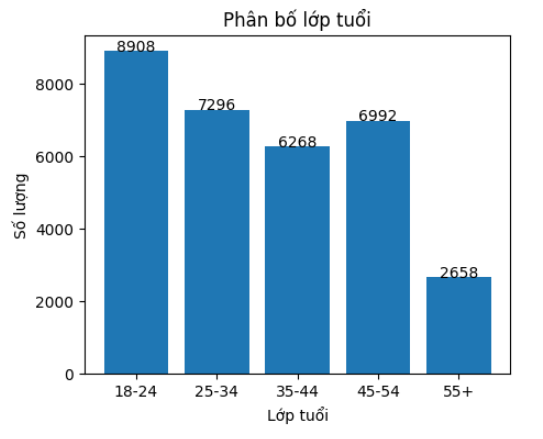
* Là dữ liệu dạng đơn biến, có 5 giá trị tương ứng với 5 nhãn.(‘18-24’, ‘25-34’, ‘35-44’, ‘45-54’, 55+).
* Các nhãn có thể được coi là một **chuỗi rời rạc** (**discrete sequence**). Do các nhãn này đại diện cho các nhóm riêng biệt và không có giá trị trung gian giữa các nhóm này.

#### **2.1.1.3. Thống kê dữ liệu**:

Do đặc thù của dữ liệu là rời rạc (có thể coi là có thứ tự tự nhiên),.. Nên ta sẽ không sử dụng các thống kê mô tả như: **trung bình**, **trung vị**, **phương sai**, **độ lệch chuẩn**,...

Thay vào đó ta trực quan hóa bằng biểu đồ phân bố tần suất, và tỉ lệ phần trăm.

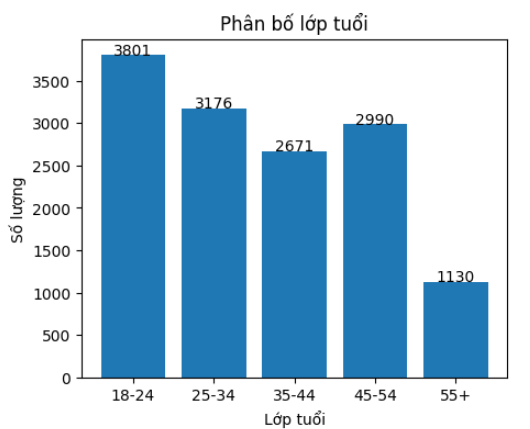
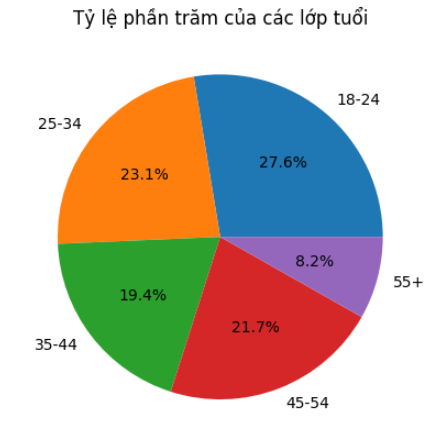
* Tập train:



|  |  |
| --- | --- |
| Hình a | Hình b |

##### *Hình 2.1. Biểu đồ tần suất (a) và tỉ lệ phần trăm tương ứng (b) của nhãn dữ liệu trên tập train*

* **Nhận xét :**
* **Có 32122 mẫu (nhãn) được cho trước.**
* **Phân bố không đồng đều:** Số lượng dữ liệu trong các nhóm tuổi không đồng đều nhau. Nhóm tuổi 18-24 có số lượng **lớn nhất** (8908 mẫu), trong khi nhóm tuổi 55+ có số lượng **nhỏ nhất** chỉ có 2658 mẫu (tương đương ⅓ so với nhóm 18-24). Điều này có thể hưởng quá trình huấn luyện mô hình trong việc dự đoán (học lệch) và **khó khăn trong việc phát hiện mẫu ít.**
* **Xu hướng giảm dần:** Số lượng dữ liệu có xu hướng giảm dần theo độ tuổi, từ nhóm tuổi trẻ nhất đến nhóm tuổi cao nhất.
* Tập test:



|  |  |
| --- | --- |
|  |  |

Hình a Hình b

##### *Hình 2.2. Biểu đồ tần suất (a) và biểu đồ phần trăm tương ứng (b) của nhãn dữ liệu trên tập test*

* **Nhận xét:**
* Sự phân bố dữ liệu trên tập test tương đối tương đồng với tập train (trên 99 %)

### **2.1.2. Dữ liệu đặc trưng (Group\_id)**

Do ta sử dụng tập train để huấn luyện và tập test để kiểm tra nên ta chỉ xét dữ liệu đặc trưng trên tập train

#### **2.1.2.1. Trên toàn bộ dữ liệu**

##### **2.1.2.1.1. Thuộc tính**

* Là các group\_id có thuộc tính rời rạc (Vô hạn đếm được).
* Thuộc kiểu dữ liệu phân loại định danh (Nominal categorical data). Những dữ liệu này sẽ không có tính thứ tự, và tương đối không liên quan gì tới nhau.

##### **2.1.2.1.2. Dạng dữ liệu**

* Là dạng dữ liệu đa biến, bao gồm các group\_id có các số lượng khác nhau ( sẽ được đề cập đến ở phần sau - thống kê).

##### **2.1.2.1.3. Thống kê dữ liệu**

1. Tổng quan

Có **331518** group\_id khác nhau xuất hiện ở tập train. Tổng số lần xuất hiện của các group\_id là: **883822.** Trong đó:

|  |  |  |
| --- | --- | --- |
| group\_id | Số lần xuất hiện | Ghi chú |
| 313318492090124 | 590 | Nhiều nhất |
| 239402 số group\_id khác nhau | 1 | Ít nhất |

###### Bảng 2.1. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất

1. Chi tiết phần thống kê lớn nhất và nhỏ nhất

# 

###### Hình 2.3. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất

**Nhận xét:**

* Trong top 80 này khoảng dao động số lần xuất hiện từ khoảng 200 đến 590.
* Chỉ có khoảng 18 nhóm có số lần xuất hiện từ 300 nhóm trở lên.

Như đã đề cập ở phần trên thì có tận 239402 là các group\_id chỉ xuất hiện 1 lần, ta không thể trực quan số group có số lần xuất hiện nhỏ nhất vì nó không có ý nghĩa. Thay vào đó, ta sẽ lập bảng đếm xem có bao nhiêu group\_id có số lần xuất hiện từ 1 đến 10.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Lần xuất hiện | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Số group\_id | 239402 | 36676 | 15215 | 8485 | 5568 | 3853 | 3072 | 2043 | 1840 | 1530 |

###### 

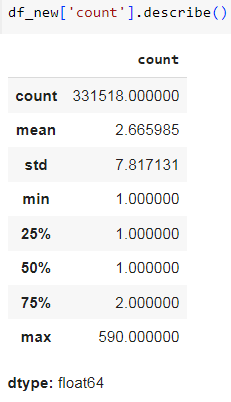
###### Bảng 2.2. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10

**Nhận xét**:

* Số lần xuất hiện ít (từ 1 đến 10) chiếm 36,3% số lần xuất hiện các nhóm. Nhưng nếu xét theo số người dùng thì chỉ dao động trong khoảng xấp xỉ 0,00003 - 0,0003. Một khoảng vô cùng nhỏ và có thể khiến cho mô hình ít cải thiện hơn dự đoán của mình và tốn thêm các tài nguyên khác khi sử dụng.

1. Chi tiết về số lần xuất hiện của các group\_id

Do group\_id gắn liền với số lần xuất hiện, ta sẽ thống kê thêm các yếu tố liên quan về số lần xuất hiện của các nhóm như: **mean**, **std**, ,.v..v



###### Hình 2.4. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id

**Nhận xét:**

* **Số lượng mẫu (count):**
* **331,518:** Đây là tổng số nhóm được thống kê.
* **Giá trị trung bình (mean):**
* **2.665985:** Trung bình mỗi nhóm xuất hiện khoảng 2.67 lần. Điều này cho thấy phần lớn các nhóm xuất hiện ít lần.
* **Độ lệch chuẩn (std):**
* **7.817131:** Độ lệch chuẩn khá cao, cho thấy sự phân tán lớn trong số lần xuất hiện của các nhóm. Một số nhóm xuất hiện rất nhiều lần, trong khi nhiều nhóm khác xuất hiện rất ít.
* **Giá trị nhỏ nhất (min):**
* **1:** Giá trị nhỏ nhất là 1, nghĩa là có nhiều nhóm chỉ xuất hiện một lần.
* **Các phần trăm vị (percentiles):**
* **25% (Q1): 1:** 25% số nhóm xuất hiện ít nhất 1 lần.
* **50% (Median): 1:** 50% số nhóm xuất hiện ít nhất 1 lần. Điều này cho thấy phần lớn các nhóm chỉ xuất hiện một lần.
* **75% (Q3): 2.:** 75% số nhóm xuất hiện ít hơn 2 lần. Điều này cho thấy rằng chỉ có 25% số nhóm xuất hiện nhiều hơn 2 lần.
* **Giá trị lớn nhất (max):**
* **590:** Giá trị lớn nhất là 590, nghĩa là có một nhóm xuất hiện tới 590 lần. Đây là một ngoại lệ và cho thấy sự chênh lệch lớn giữa các nhóm.

#### **2.1.2.2. Trên các lớp dữ liệu.**

Các đặc điểm như thuộc tính hay dạng dữ liệu ta đã đề cập ở phần trên, nên ta sẽ không nói lại ở phần này.

##### **2.1.2.2.1. Lớp ‘18-24’**

1. **Tổng quan**

* Có **142008** group\_id khác nhau xuất hiện ở tập train. Tổng số lần xuất hiện của các group\_id là: **301815.** Trong đó:

|  |  |  |
| --- | --- | --- |
| Group id | Số lần xuất hiện | Ghi chú |
| 1173636692750000 | 453 | Nhiều nhất |
| 107403 số group\_id khác nhau | 1 | Ít nhất |

###### Bảng 2.3. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất lớp ‘18-24’

1. **Chi tiết về lớn nhất và nhỏ nhất**

###### Hình 2.5. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất lớp ‘18-24’

**Nhận xét:**

* Trong top 80 này khoảng dao động số lần xuất hiện từ khoảng 89 đến 453.
* Chỉ có khoảng 6 nhóm là có số lần xuất hiện từ 200 nhóm trở lên.

Như đã đề cập ở phần trên thì có tận 107403 là các group\_id chỉ xuất hiện 1 lần, ta không thể trực quan số group có số lần xuất hiện nhỏ nhất vì nó không có ý nghĩa. Thay vào đó, ta sẽ lập bảng đếm xem có bao nhiêu group\_id có số lần xuất hiện từ 1 đến 10.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Số lần xuất hiện** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **Số group\_id** | **107403** | **14886** | **5931** | **3275** | **2132** | **1492** | **1182** | **856** | **683** | **523** |

###### Bảng 2.4. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10

**Nhận xét**:

* Số lần xuất hiện ít (từ 1 đến 10) chiếm 45.8% số lần xuất hiện các nhóm.

1. **Chi tiết về số lần xuất hiện của các group\_id**

###### Hình 2.6. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id lớp ‘18-24’

**Nhận xét:**

* **Số lượng mẫu (count):**
* 142,008: Đây là tổng số nhóm được thống kê.
* **Giá trị trung bình (mean):**
* **2.1253:** Trung bình mỗi nhóm xuất hiện khoảng 2.13 lần. Điều này cho thấy phần lớn các nhóm xuất hiện ít lần.
* **Độ lệch chuẩn (std):**
* **5.2751:** Độ lệch chuẩn khá cao, cho thấy sự phân tán lớn trong số lần xuất hiện của các nhóm. Một số nhóm xuất hiện rất nhiều lần, trong khi nhiều nhóm khác xuất hiện rất ít.
* **Giá trị nhỏ nhất (min):**
* **1:** Giá trị nhỏ nhất là 1, nghĩa là có nhiều nhóm chỉ xuất hiện một lần.
* **Các phần trăm vị (percentiles):**
* **25% (Q1): 1:** 25% số nhóm xuất hiện ít nhất 1 lần.
* **50% (Median): 1:** 50% số nhóm xuất hiện ít nhất 1 lần. Điều này cho thấy phần lớn các nhóm chỉ xuất hiện một lần.
* **75% (Q3): 1:** 75% số nhóm xuất hiện ít nhất 1 lần. Điều này cho thấy rằng chỉ có 25% số nhóm xuất hiện nhiều hơn 1 lần.
* **Giá trị lớn nhất (max):**
* **453:** Giá trị lớn nhất là 453, nghĩa là có một nhóm xuất hiện tới 453 lần. Đây là một ngoại lệ và cho thấy sự chênh lệch lớn giữa các nhóm.

##### 

##### **2.1.2.2.2. Lớp ’25-34’**

1. **Tổng quan**

* Có **122747** group\_id khác nhau xuất hiện ở tập train. Tổng số lần xuất hiện của các group\_id là: **264910.** Trong đó:

|  |  |  |
| --- | --- | --- |
| Group id | Số lần xuất hiện | Ghi chú |
| 313318492090124 | 205 | Nhiều nhất |
| 89713 số group\_id khác nhau | 1 | Ít nhất |

###### Bảng 2.5. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất lớp ‘25-34’

1. **Chi tiết về lớn nhất và nhỏ nhất**

###### Hình 2.7. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất lớp ‘25-34’

**Nhận xét:**

* Trong top 80 này khoảng dao động số lần xuất hiện từ khoảng 69 đến 205.
* Chỉ có khoảng 26 nhóm là có số lần xuất hiện từ 100 nhóm trở lên.

Như đã đề cập ở phần trên thì có tận 89713 là các group\_id chỉ xuất hiện 1 lần, ta không thể trực quan số group có số lần xuất hiện nhỏ nhất vì nó không có ý nghĩa. Thay vào đó, ta sẽ lập bảng đếm xem có bao nhiêu group\_id có số lần xuất hiện từ 1 đến 10.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Số lần xuất hiện** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **Số group\_id** | **89713** | **13617** | **5781** | **3373** | **2109** | **1545** | **1075** | **929** | **696** | **589** |

###### Bảng 2.6. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10

**Nhận xét**:

* Số lần xuất hiện ít (từ 1 đến 10) chiếm 45.1% số lần xuất hiện các nhóm.

1. **Chi tiết về số lần xuất hiện của các group\_id**

###### Hình 2.8. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id lớp ‘25-34’

**Nhận xét:**

* **Số lượng mẫu (count):**
* **122,747:** Đây là tổng số nhóm được thống kê.
* **Giá trị trung bình (mean):**
* **2.1581:** Trung bình mỗi nhóm xuất hiện khoảng 2.16 lần. Điều này cho thấy phần lớn các nhóm xuất hiện ít lần.
* **Độ lệch chuẩn (std):**
* **4.4273:** Độ lệch chuẩn nhỏ hơn lớp ‘18-24’ nhưng vẫn khá cao, cho thấy sự phân tán lớn trong số lần xuất hiện của các nhóm. Một số nhóm xuất hiện rất nhiều lần, trong khi nhiều nhóm khác xuất hiện rất ít.
* **Giá trị nhỏ nhất (min):**
* **1:** Giá trị nhỏ nhất là 1, nghĩa là có nhiều nhóm chỉ xuất hiện một lần.
* **Các phần trăm vị (percentiles):**
* **25% (Q1): 1:** 25% số nhóm xuất hiện ít nhất 1 lần.
* **50% (Median): 1:** 50% số nhóm xuất hiện ít nhất 1 lần. Điều này cho thấy phần lớn các nhóm chỉ xuất hiện một lần.
* **75% (Q3): 2:** 75% số nhóm xuất hiện ít hơn 2 lần. Điều này cho thấy rằng chỉ có 25% số nhóm xuất hiện nhiều hơn 2 lần.
* **Giá trị lớn nhất (max):**
* **205:** Giá trị lớn nhất là 205, nghĩa là có một nhóm xuất hiện tới 205 lần. Đây là một ngoại lệ và cho thấy sự chênh lệch lớn giữa các nhóm.

##### **2.1.2.2.3. Lớp ‘35-44’**

1. **Tổng quan**

* Có **96288** group\_id khác nhau xuất hiện ở tập train. Tổng số lần xuất hiện của các group\_id là: **183109.** Trong đó:

|  |  |  |
| --- | --- | --- |
| Group id | Số lần xuất hiện | Ghi chú |
| 184730418261517 | 196 | Nhiều nhất |
| 70795 số group\_id khác nhau | 1 | Ít nhất |

###### Bảng 2.7. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất lớp ‘35-44’

1. **Chi tiết về lớn nhất và nhỏ nhất**

###### Hình 2.9. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất lớp ‘35-44’

**Nhận xét:**

* Trong top 80 này khoảng dao động số lần xuất hiện từ khoảng 43 đến 196.
* Chỉ có khoảng 4 nhóm là có số lần xuất hiện từ 100 nhóm trở lên.

Như đã đề cập ở phần trên thì có tận 70795 là các group\_id chỉ xuất hiện 1 lần, ta không thể trực quan số group có số lần xuất hiện nhỏ nhất vì nó không có ý nghĩa. Thay vào đó, ta sẽ lập bảng đếm xem có bao nhiêu group\_id có số lần xuất hiện từ 1 đến 10.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Số lần xuất hiện** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **Số group\_id** | **70795** | **11464** | **4755** | **2672** | **1662** | **1162** | **792** | **533** | **443** | **355** |

###### Bảng 2.8. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10

**Nhận xét**:

* Số lần xuất hiện ít (từ 1 đến 10) chiếm 51.7% số lần xuất hiện các nhóm.

1. **Chi tiết về số lần xuất hiện của các group\_id**

###### Hình 2.10. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id lớp ‘35-44’

**Nhận xét:**

* **Số lượng mẫu (count):**
* 96,288: Đây là tổng số nhóm được thống kê của lớp ‘35-44’.
* **Giá trị trung bình (mean):**
* **1.9017:** Trung bình mỗi nhóm xuất hiện khoảng 1.90 lần. Điều này cho thấy phần lớn các nhóm xuất hiện ít lần.
* **Độ lệch chuẩn (std):**
* **3.1774:** Độ lệch chuẩn nhỏ hơn lớp ‘25-34’ nhưng vẫn khá cao, cho thấy sự phân tán lớn trong số lần xuất hiện của các nhóm. Một số nhóm xuất hiện rất nhiều lần, trong khi nhiều nhóm khác xuất hiện rất ít.
* **Giá trị nhỏ nhất (min):**
* **1:** Giá trị nhỏ nhất là 1, nghĩa là có nhiều nhóm chỉ xuất hiện một lần.
* **Các phần trăm vị (percentiles):**
* **25% (Q1): 1:** 25% số nhóm xuất hiện ít nhất 1 lần.
* **50% (Median): 1:** 50% số nhóm xuất hiện ít nhất 1 lần. Điều này cho thấy phần lớn các nhóm chỉ xuất hiện một lần.
* **75% (Q3): 2:** 75% số nhóm xuất hiện ít hơn 2 lần. Điều này cho thấy rằng chỉ có 25% số nhóm xuất hiện nhiều hơn 2 lần.
* **Giá trị lớn nhất (max):**
* **196:** Giá trị lớn nhất là 196, nghĩa là có một nhóm xuất hiện tới 196 lần. Đây là một ngoại lệ và cho thấy sự chênh lệch lớn giữa các nhóm.

##### **2.1.2.2.3. Lớp ‘45-54’**

1. **Tổng quan**

* Có **59686** group\_id khác nhau xuất hiện ở tập train. Tổng số lần xuất hiện của các group\_id là: **107129.** Trong đó:

|  |  |  |
| --- | --- | --- |
| Group id | Số lần xuất hiện | Ghi chú |
| 1650750468512080 | 173 | Nhiều nhất |
| 45724 số group\_id khác nhau | 1 | Ít nhất |

###### Bảng 2.9. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất lớp ‘45-54’

1. **Chi tiết về lớn nhất và nhỏ nhất**

###### Hình 2.11. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất lớp ‘45-54’

**Nhận xét:**

* Trong top 80 này khoảng dao động số lần xuất hiện từ khoảng 34 đến 173.
* Chỉ có khoảng 5 nhóm là có số lần xuất hiện từ 100 nhóm trở lên.

Như đã đề cập ở phần trên thì có tận 45724 là các group\_id chỉ xuất hiện 1 lần, ta không thể trực quan số group có số lần xuất hiện nhỏ nhất vì nó không có ý nghĩa. Thay vào đó, ta sẽ lập bảng đếm xem có bao nhiêu group\_id có số lần xuất hiện từ 1 đến 10.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Số lần xuất hiện** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **Số group\_id** | **45725** | **6491** | **2557** | **1419** | **841** | **595** | **413** | **322** | **219** | **188** |

###### Bảng 2.10. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10

**Nhận xét**:

* Số lần xuất hiện ít (từ 1 đến 10) chiếm 54.9% số lần xuất hiện các nhóm.

1. **Chi tiết về số lần xuất hiện của các group\_id**

###### Hình 2.12. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id lớp ‘45-54’

**Nhận xét:**

* **Số lượng mẫu (count):**
* **59,686:** Đây là tổng số nhóm được thống kê của lớp ‘45-54’.
* **Giá trị trung bình (mean):**
* **1.7948:** Trung bình mỗi nhóm xuất hiện khoảng 1.79 lần. Điều này cho thấy phần lớn các nhóm xuất hiện ít lần.
* **Độ lệch chuẩn (std):**
* **3.1314:** Độ lệch chuẩn nhỏ hơn lớp ‘35-44’ nhưng vẫn khá cao, cho thấy sự phân tán lớn trong số lần xuất hiện của các nhóm. Một số nhóm xuất hiện rất nhiều lần, trong khi nhiều nhóm khác xuất hiện rất ít.
* **Giá trị nhỏ nhất (min):**
* **1:** Giá trị nhỏ nhất là 1, nghĩa là có nhiều nhóm chỉ xuất hiện một lần.
* **Các phần trăm vị (percentiles):**
* **25% (Q1): 1:** 25% số nhóm xuất hiện ít nhất 1 lần.
* **50% (Median): 1:** 50% số nhóm xuất hiện ít nhất 1 lần. Điều này cho thấy phần lớn các nhóm chỉ xuất hiện một lần.
* **75% (Q3): 1:** 75% số nhóm xuất hiện ít khoảng 1 lần. Điều này cho thấy rằng chỉ có 25% số nhóm xuất hiện nhiều hơn 1 lần.
* **Giá trị lớn nhất (max):**
* **173:** Giá trị lớn nhất là 173, nghĩa là có một nhóm xuất hiện tới 173 lần. Đây là một ngoại lệ và cho thấy sự chênh lệch lớn giữa các nhóm.

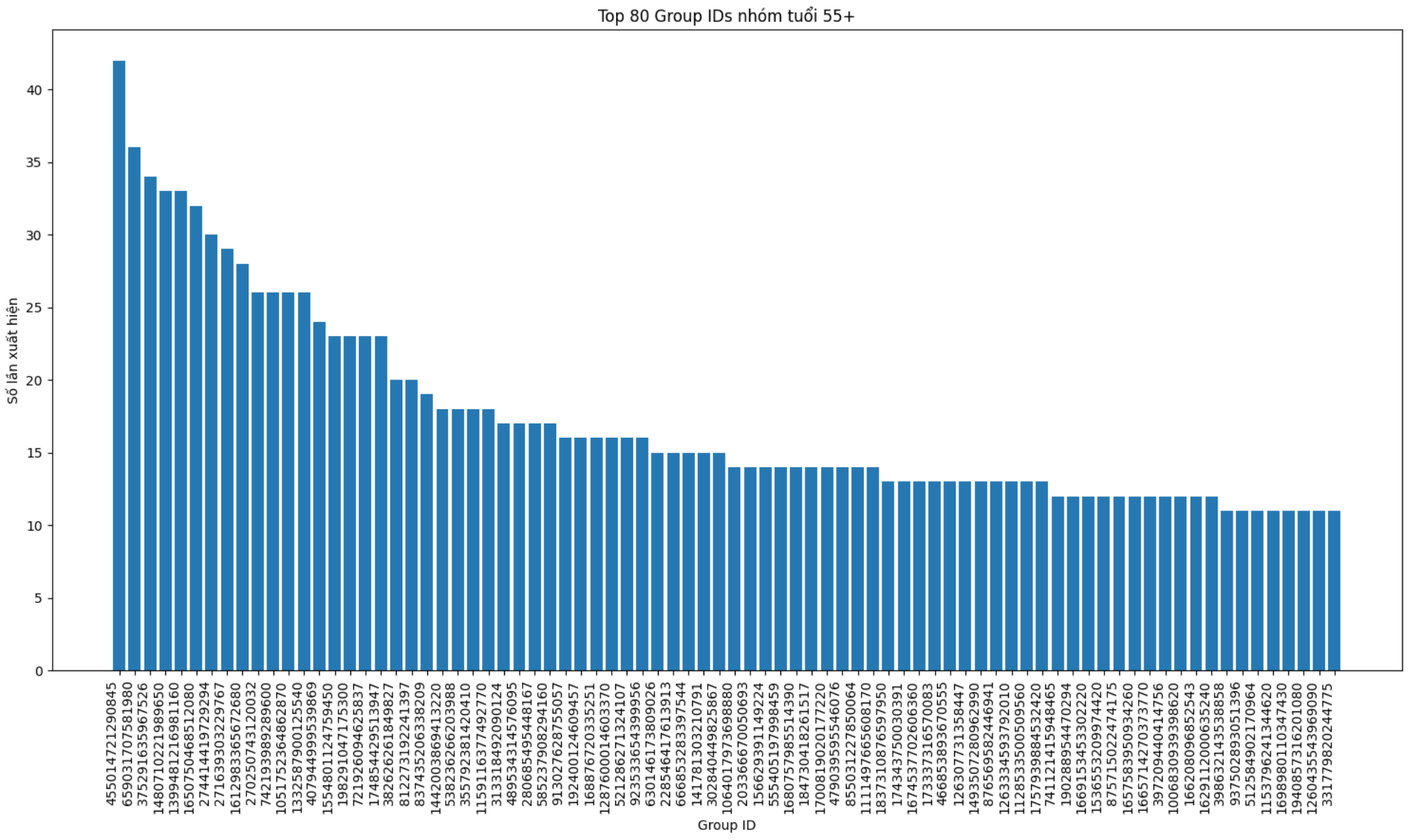
##### **2.1.2.2.4. Lớp ‘55+’**

1. **Tổng quan**

* Có **19543** group\_id khác nhau xuất hiện ở tập train. Tổng số lần xuất hiện của các group\_id là: **26859.** Trong đó:

|  |  |  |
| --- | --- | --- |
| Group id | Số lần xuất hiện | Ghi chú |
| 455014721290845 | 42 | Nhiều nhất |
| 16296 số group\_id khác nhau | 1 | Ít nhất |

###### Bảng 2.11. Bảng thống kê group\_id xuất hiện nhiều nhất và ít nhất lớp ‘55+’

1. **Chi tiết về lớn nhất và nhỏ nhất**

###### Hình 2.13. Biểu đồ top 80 group\_id có số lần xuất hiện nhiều nhất lớp ‘55+’

**Nhận xét:**

* Trong top 80 này khoảng dao động số lần xuất hiện từ khoảng 11 đến 42.
* Chỉ có khoảng 19 nhóm là có số lần xuất hiện từ 20 nhóm trở lên.

Như đã đề cập ở phần trên thì có tận 16296 là các group\_id chỉ xuất hiện 1 lần, ta không thể trực quan số group có số lần xuất hiện nhỏ nhất vì nó không có ý nghĩa. Thay vào đó, ta sẽ lập bảng đếm xem có bao nhiêu group\_id có số lần xuất hiện từ 1 đến 10.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Số lần xuất hiện** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **Số group\_id** | **16296** | **1860** | **662** | **278** | **145** | **85** | **57** | **30** | **25** | **19** |

###### Bảng 2.12. Bảng thống kê số lượng nhóm tương ứng với số lần xuất hiện ít nhất từ 1 đến 10

**Nhận xét**:

* Số lần xuất hiện ít (từ 1 đến 10) chiếm 72.4% số lần xuất hiện các nhóm.

1. **Chi tiết về số lần xuất hiện của các group\_id**

###### Hình 2.14. Các thuộc tính liên quan đến số lần xuất hiện của các group\_id lớp ‘55+’

**Nhận xét:**

* **Số lượng mẫu (count):**
* **19,543**: Đây là tổng số nhóm được thống kê của lớp ‘55+’.
* **Giá trị trung bình (mean):**
* **1.3746:** Trung bình mỗi nhóm xuất hiện khoảng 1.37 lần. Điều này cho thấy phần lớn các nhóm xuất hiện ít lần.
* **Độ lệch chuẩn (std):**
* **1.4530:** Độ lệch chuẩn nhỏ hơn lớp ‘35-44’ , cho thấy sự phân tán không quá lớn trong số lần xuất hiện của các nhóm. Một số nhóm xuất hiện rất nhiều lần, trong khi nhiều nhóm khác xuất hiện rất ít.
* **Giá trị nhỏ nhất (min):**
* **1:** Giá trị nhỏ nhất là 1, nghĩa là có nhiều nhóm chỉ xuất hiện một lần.
* **Các phần trăm vị (percentiles):**
* **25% (Q1): 1:** 25% số nhóm xuất hiện ít nhất 1 lần.
* **50% (Median): 1:** 50% số nhóm xuất hiện ít nhất 1 lần. Điều này cho thấy phần lớn các nhóm chỉ xuất hiện một lần.
* **75% (Q3): 1:** 75% số nhóm xuất hiện ít khoảng 1 lần. Điều này cho thấy rằng chỉ có 25% số nhóm xuất hiện nhiều hơn 1 lần.
* **Giá trị lớn nhất (max):**
* **42:** Giá trị lớn nhất là 42, nghĩa là có một nhóm xuất hiện tới 42 lần. Đây là một ngoại lệ và cho thấy sự chênh lệch lớn giữa các nhóm.

# 

# 

# 

# **PHẦN 3: TIỀN XỬ LÝ DỮ LIỆU VÀ BIỂU DIỄN DỮ LIỆU**

## 

## **3.1. Trên toàn bộ dữ liệu**

### **3.1.1. Làm sạch dữ liệu**

#### **3.1.1.1. Đánh giá các yếu tố**

* **Các giá trị bị thiếu**: Do ta xét các danh sách group\_id trên toàn bộ tập dữ liệu nên việc thiếu dữ liệu ta có thể coi là không có.
* **Các giá trị của group\_id:** được cho ban đầu có thể coi là đúng (có giá trị chính xác) và nhất quán.
* **Các giá trị nhiễu**: Như đã thống kê ở phần trước (2.1.2.1) thì ta có thể thấy các giá trị xuất hiện ít nhất( từ 1 đến 10, hoặc nhiều hơn) và các nhóm có số lần xuất hiện nhiều nhất chênh lệch với trung bình ( > 390) có khoảng 8 nhóm.

#### **3.1.1.2. Cách xử lý**

* **Loại bỏ các giá trị nhiễu**:

Ta sẽ lựa chọn hướng giải pháp là phương pháp **IQR** . Trong đó **IQR** = **Q3 Q1**, với **Q3** (**Quartile 3**: Phân vị thứ ba, tức là giá trị tại **75%** của dữ liệu.) và **Q1** (**Quartile 1**: Phân vị thứ nhất, tức là giá trị tại **25%** của dữ liệu).

Giới hạn dưới: **Q1 - 1,5 IQR**

Giới hạn trên: **Q3 + 1.5 IQR**

Giá trị nào **nhỏ hơn** **giới hạn dưới** và **lớn hơn giới hạn trên** được coi là ngoại lai và được loại bỏ.

|  |
| --- |
| Hình 3.1. Đoạn mã sử dụng phương pháp IQR |

* **Sử dụng mô hình:**

Ta sẽ sử dụng mô hình hồi quy Logistic Regression để phát hiện và loại bỏ điểm ngoại lai bằng phần dư. Các bước thực hiện bao gồm:

* **Dự đoán trên Tập Huấn Luyện**: Sử dụng mô hình đã huấn luyện để dự đoán nhãn cho tập dữ liệu huấn luyện.
* **Tính Toán Phần Dư**: Tính toán phần dư bằng cách lấy hiệu giữa nhãn thực tế và nhãn dự đoán từ mô hình.
* **Xác Định Ngưỡng Loại Bỏ Điểm Ngoại Lai**: Xác định ngưỡng để loại bỏ điểm ngoại lai bằng cách sử dụng 2 lần độ lệch chuẩn của phần dư. Điểm dữ liệu có phần dư lớn hơn ngưỡng sẽ được coi là điểm ngoại lai.
* **Loại Bỏ Điểm Ngoại Lai**: Loại bỏ các điểm dữ liệu có phần dư lớn hơn ngưỡng, và tạo ra tập dữ liệu đã được làm sạch.
* **Mục tiêu**: Phương pháp này giúp loại bỏ các điểm dữ liệu bất thường hoặc không phù hợp, nhằm cải thiện hiệu suất của mô hình bằng cách chỉ giữ lại những điểm dữ liệu phù hợp.

|  |
| --- |
| Hình 3.2. Đoạn mã sử dụng phương pháp dùng mô hình để loại bỏ điểm ngoại lai |

Như vậy, ta sẽ coi có các vấn đề như sau và chúng sẽ được đánh giá theo các vấn đề ở phần sau:

##### *Vấn đề 1: Loại bỏ các giá trị nhiễu bằng phương pháp IQR*

##### *Vấn đề 2: Sử dụng mô hình logistic Regression để dự đoán ra các điểm ngoại lại và bỏ các giá trị đó*

### **3.1.2. Cân bằng và tích hợp dữ liệu**

* **Cân bằng**:

Ta dựa trên thống kê ở phần trước để cân bằng 2 lớp ‘18-24’ và ‘55+’ theo giải pháp: **Cân bằng dữ liệu (Data Balancing)**

* **Oversampling (thêm các mẫu)**: Các số lượng ít: Mẫu 55+ bằng cách viết hàm  **generate\_samples** (top\_groups, probabilities, num\_samples, num\_groups) để sinh ra 3000 mẫu mới dựa vào xác suất xuất hiện của group\_id tốt nhất của mẫu số 5 (lớp 4) và số lượng nhóm được xác định dựa trên xác suất số lượng nhóm với mô tả:
  + Chuyển đổi xác suất thành danh sách và chuẩn hóa: Xác suất của các nhóm được chuyển đổi thành một mảng và chuẩn hóa để đảm bảo tổng xác suất bằng 1.
  + Điều chỉnh xác suất còn thiếu: Nếu tổng xác suất ban đầu nhỏ hơn 1, phần xác suất còn thiếu sẽ được chia đều cho các nhóm để đảm bảo tổng xác suất bằng 1.
  + Sinh các mẫu dữ liệu: Sử dụng phân phối Poisson để xác định số lượng nhóm trong mỗi mẫu.
  + Chọn ngẫu nhiên các nhóm dựa trên xác suất đã chuẩn hóa và thêm vào danh sách các mẫu.
  + Trả về danh sách các mẫu: Danh sách các mẫu được trả về dưới dạng danh sách các danh sách con, mỗi danh sách con chứa các nhóm được chọn cho.

|  |
| --- |
| Hình 3.3. Hàm sinh mẫu ngẫu nhiên dựa trên xác suất |

* **Sử dụng thư viện SMOTE:** Để cân bằng các mẫu.
* **Tích hợp**: Ở trong bài toán này dữ liệu của ta đã rõ ràng, và ta có thể bỏ qua bước tích hợp dữ liệu.

### **3.1.3. Biểu diễn dữ liệu.**

#### **3.1.3.1. Nhãn (Label)**:

Từ việc xác định thuộc tính rời rạc ta sẽ tạo ra các giá trị đại diện cho các lớp thành các con số (label encoding) đại diện như sau:



###### Hình 3.4. Các giá trị đại diện của label

#### **3.1.3.2. Các đặc trưng từ group\_id**

##### **3.1.3.2.1. Sử dụng thư viện có sẵn để trích xuất ra đặc trưng**

Sử dụng lớp CountVectorizer để chuyển đổi các Group\_id mà người dùng tham gia (hiện đang ở dạng strings) thành các vector số học, với mỗi phần tử của vector là một group\_id cụ thể. Điều này giúp mô hình có thể coi các group\_id là những đặc trưng để xử lý chúng.

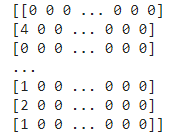
|  |
| --- |
|  |
| Hình 3.5.Đoạn mã xử lý các group\_id thành vector đặc trưng |

##### **3.1.3.2.2. Lựa chọn danh sách group\_id làm dữ liệu phân loại nhị phân**

Ở đây ta sẽ lựa chọn ra một danh sách các group\_id mà ta xử lý xong ở các bước trước làm các cột đặc trưng đối với các nhãn. Nếu người đó có tham gia cột nào thì ta sẽ thêm 1, ngược lại ta sẽ thêm 0. Các danh sách sẽ được lọc theo điều kiện của số lượng xuất hiện. Ví dụ ta sẽ lựa chọn 1 danh sách mà các group\_id có số lần xuất hiện trong khoảng từ n1 đến n2 làm đặc trưng.

|  |  |
| --- | --- |
|  |  |

###### Hình 3.6. Đoạn mã xử lý đặc trưng



###### Hình 3.7. Minh họa ma trận sau khi đã biểu diễn với cột đầu tiên là nhãn

##### **3.1.3.2.3. Các đặc trưng mới**

Ta sẽ đề cập đến các vấn đề và đánh giá, trình bày chúng ở phần sau (phần cải tiến) nếu như các đặc trưng này không đáp ứng được đầu ra của bài toán:

###### Vấn đề 3: Tạo dữ liệu đặc trưng từ số lượng nhóm tham gia của 1 người dùng.

* **Hướng 1**: Lọc ra và loại bỏ những dữ liệu gây nhiễu. Ở đây coi những dữ liệu gây nhiễu là những người dùng tham gia trên 200 nhóm (outlier) và các nhóm chỉ có 1 người tham gia.
* **Hướng 2**: Ta sẽ lựa chọn ngưỡng để xác định ranh giới.

|  |
| --- |
|  |

*Hình 3.9 Thông tin thống kê số lượng nhóm mà người dùng tham gia*

.

###### Vấn đề 4: Tạo dữ liệu đặc trưng từ các nhóm trùng nhau giữa các lớp.

|  |
| --- |
| *Hình 3.10. Ma trận biểu diễn số lượng nhóm trùng nhau trong 1500 nhóm tốt nhất của mỗi lớp* |

Dựa vào số lượng trùng nhau giữa các biến ta có các giải pháp để thay đổi đặc trưng:

1. Loại bỏ các group\_id trùng nhau giữa 2 lớp gần nhau (mở rộng ra là 3 lớp, 4 lớp, 5 lớp) theo một số lượng nhất định.
2. Ta sẽ tạo ra đặc trưng riêng phân biệt giữa các lớp bằng cách chọn ra n nhóm tốt nhất của lớp đó. Xong rồi ta lấy n1 nhóm random từ tổ hợp chập 2 (3) của n làm đặc trưng mới.
3. Kết hợp của cả 1 và 2. Ta sẽ loại bỏ các nhóm trùng nhau và từ các nhóm trùng nhau đó ta sẽ làm tương tự bước 2 để tạo ra đặc trưng mới

###### Vấn đề 5:Tạo dữ liệu đặc trưng từ độ dài của group\_id

|  |  |
| --- | --- |
| Toàn bộ dữ liệu |  |
| ‘18-24’ |  |
| ‘25-34’ |  |
| ‘35-44’ |  |
| ‘45-54’ |  |
| ‘55+’ |  |

*Bảng 3.1. Độ dài của từng nhóm xét theo toàn bộ và riêng từng lớp*

Ở đây ta sẽ tạo thêm các cột mới ở trường với tên cột là độ dài nhóm, giá trị là số lượng nhóm có độ dài tương ứng với cột.Sau đó dùng thư viện MinMaxScaler để chuẩn hóa 6 cột mới này các đặc trưng.

###### Vấn đề 6: Tạo ra các luật (rulesets) để dùng làm đặc trưng dự đoán cho mô hình sử dụng CountVectorizer làm đặc trưng

|  |
| --- |
|  |

*Hình 3.11.Định nghĩa ruleset*

###### 

## **3.2. Trên các lớp dữ liệu**

Ở đây, thay vì sử dụng toàn bộ dữ liệu, thì ta sử dụng từng dữ liệu thuộc các lớp để làm sạch dữ liệu, cân bằng dữ liệu tương tự như các công việc đã nói ở phần trên. Phần này ta sẽ chỉ đề cập đến những công đoạn được cho là khác với phần trên.

### **3.2.1. Biểu diễn dữ liệu**

#### **3.2.1.1. Các đặc trưng từ group\_id**

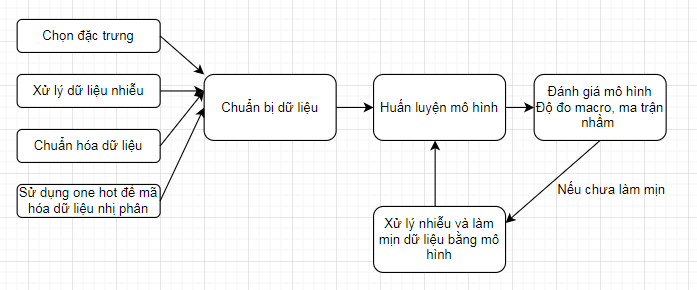
Cũng tương như trên toàn bộ dữ liệu, thì ta sẽ lựa chọn một danh sách các group\_id làm đặc trưng nhưng thay vì lấy theo số lần xuất hiện, thì ta sẽ xét riêng các lớp, sau đó ta sẽ lựa chọn ra n group\_id xuất hiện ở lớp đó làm đặc trưng của lớp đó. Cuối cùng ta tổng hợp lại được danh sách đặc trưng và làm tương tự như phần trên.

|  |
| --- |
| Hình 3.12. Đoạn mã lấy n phần tử xuất hiện nhiều nhất ở trong một lớp |

**Nhận xét**: Khi khoảng n1 và n2 càng rộng và n càng lớn thì ta được ma trận đặc trưng có độ tương đồng cao (trên 96%).

# **PHẦN 4: ĐÁNH GIÁ CÁC MÔ HÌNH**

Ta sẽ lựa chọn 1 mô hình để tinh chỉnh ra đầu vào tốt nhất. Mô hình được chọn (Logistic Regression) sẽ được trình bày chi tiết các kết quả sau mỗi lần tinh chỉnh lớn. Các mô hình còn lại sẽ được so sánh dựa trên đầu vào của phiên bản tốt nhất của mô hình Logistic Regression. Các mô hình sẽ được triển khai theo kiến trúc sau:



###### Hình 4.1. Kiến trúc mô hình giải quyết bài toán

## **4.1. Mô hình Logistic Regression**

### **4.1.1. Khi sử dụng đặc trưng trích xuất từ thư viện**

Các kĩ thuật sử dụng:

* **Class weighting**: Bởi vì có sự mất cân bằng trong tập dữ liệu mà chúng ta đang xét, việc sử dụng trọng số cho các nhãn tuổi là cần thiết để đưa ra kết quả chính xác hơn. Với công thức:
* **wj = n\_samples / (n\_classes \* n\_samplesj)**

(Với: **wj** là trọng số của từng nhãn

**n\_samples** là tổng số nhãn (người dùng) của tập dữ liệu

**n\_classesis** số các nhãn khác nhau (5 nhãn tuổi) của tập dữ liệu

**n\_samplesj** là số nhãn có label (một nhóm tuổi nhất định) mà ta đang xét)

* **K-fold cross validation:**
* Ta chia tập dữ liệu train thành k phần, và với mỗi phần, ta lại:
* Sử dụng nhóm hiện tại để đánh giá hiệu quả mô hình
* Các nhóm còn lại được sử dụng để huấn luyện mô hình
* Huấn luyện mô hình
* Đánh giá và sau đó hủy mô hình

|  |
| --- |
| Hình 4.2. Đoạn mã sử dụng classWeight và k-fold cho mô hình |
| |  |  | | --- | --- | |  |  | |
| Tập train Tập test |

###### Hình 4.3. Kết quả của mô hình khi sử dụng cross validation với k = 5 và class weight trên tập train và tập test

|  |
| --- |
|  |

###### Hình 4.4. Ma trận nhầm lẫn tương ứng

**Nhận xét:**

* Ta dễ dàng nhận ra sự chênh lệch giữa 2 tập khi dùng mô hình test cũng không chênh lệch lớn và đạt được độ chấp nhận được là 0.81.
* Nhãn tuổi 18-24 và 45-54 có tỉ lệ recall tốt nhất (0.72 và 0.68).
* Nhãn tuổi 25-35 và 35-44 có tỉ lệ recall trung bình (0.49 và 0.43).
* Nhãn tuổi 55+ có tỉ lệ recall thấp (chỉ 0.18).
* Mặc dù như vậy, nhưng nếu ta xét trên ma trận nhầm lẫn, ta có thể thấy rằng mô hình có khả năng đoán gần đúng nhãn tuổi khá tốt (0.83).
* Mô hình thường xuyên bị nhầm nhãn tuổi 45-54 với 4 nhóm tuổi còn lại.
* Mô hình đoán lớp 0 tốt nhất và lớp 4 tệ nhất.

=> Có thể thấy, mô hình đánh giá khá tốt ở mức từ chấp nhận được đến tốt ở 3 nhãn tuổi đầu tiên, do có nguồn dữ liệu đủ. Còn ở nhãn tuổi 45-54, model dễ bị nhầm lẫn và đoán sang các nhóm tuổi khác hơn, cho thấy rằng có lẽ nhóm tuổi này có nhiễu. Và ở nhóm tuổi 55+, độ chính xác của mô hình có thể nói là trung bình thấp, mặc dù đã thêm trọng số cho nhãn tuổi này, nhưng có lẽ do tập dữ liệu còn ít nên mô hình cũng chưa thể học được nhiều về nhóm tuổi này để dự đoán chính xác nó.

### **4.1.2. Khi sử dụng đặc trưng theo cách tiếp cận lấy từ hàm matrix feature**

Ở đây ta sẽ tối ưu lần lượt các giải pháp và vấn đề thay vì tối ưu chúng một cách đồng thời.

#### **4.1.2.1. Chưa đề cập đến các vấn đề**

* Khi ta chọn **n1 = 50**, **n2 = 390** thì danh sách nhóm đặc trưng (của tất cả các lớp với số lần xuất hiện lớn hơn 50 và nhỏ hơn 390) sẽ có **1388 nhóm**. Nên ta sẽ chọn ra **n = 278** (nhóm đặc trưng của mỗi lớp) và ta có **1390 nhóm** đặc trưng.

|  |  |
| --- | --- |
| 1388 nhóm | 1390 nhóm |
|  |  |
|  |  |

###### Bảng 4.1. So sánh giữa 2 cách lấy số nhóm làm đặc trưng với số lượng nhóm trung bình

**Nhận xét**:

* Với số lượng nhóm đặc trưng vào khoảng 1390 nhóm thì ta thấy lựa chọn số nhóm theo số lần xuất hiện trên toàn bộ các nhóm thì việc mô hình phân loại các lớp 0, 1, 2 tốt hơn nhưng lại đoán nhầm các lớp khác cũng nhiều hơn. Ngược lại, với dữ liệu khi phân chia ra lựa chọn các nhóm tốt nhất theo từng lớp thì mô hình đoán các lớp khác ít hơn và cả 2 có số lượng đoán nhầm vào lớp 4 cao.

Như đã đề cập ở phần trước thì khi tăng số lượng nhóm theo 2 cách thì 2 danh sách nhóm đặc trưng sẽ tương đồng với nhau nên ta sẽ chỉ xét 1 cách và xem mô hình có cải thiện hơn khi ta thêm đặc trưng vào hay không. Ta sẽ xét cách lựa chọn theo nhóm với n = 2000 ta sẽ có 6410 nhóm khác nhau làm đặc trưng và so sánh kết quả trên tập train với tập test:

|  |  |
| --- | --- |
| Tập test | Tập train |
|  |  |
|  |  |

###### Bảng 4.2. So sánh hiệu quả mô hình trên tập train và tập test

**Nhận xét:**

* **Độ chênh lệch** **kết quả**: Chênh lệch 19% đối với accuracy và 7-8 % đối với độ chấp nhận được cho thấy mô hình đang gặp vấn đề với việc tổng quát.
* Với tập test khi tăng nhóm đặc trưng lên:
* **Lớp 0.0** có hiệu suất tốt nhất với **precision** là **0.75** và **f1-score** là **0.70**, cho thấy mô hình dự đoán khá chính xác cho lớp này.
* **Lớp 1.0** và **lớp 2.0** có **precision** và **recall** thấp hơn, cho thấy mô hình gặp khó khăn trong việc phân loại chính xác các lớp này.
* **Lớp 3.0** có **recall cao (0.69)** nhưng **precision** thấp **(0.37)**, cho thấy mô hình có xu hướng dự đoán quá nhiều mẫu thuộc lớp này.
* **Lớp 4.0** có hiệu suất kém nhất với **precision** là **0.4** và **f1-score** là **0.2**, cho thấy mô hình cần cải thiện đáng kể để dự đoán chính xác lớp này.
* **Độ chính xác** và **độ chấp nhận** được của mô hình đều tăng lên, như vậy khi tăng số lượng nhóm đặc trưng lên thì mô hình cải thiện hơn.
* Mô hình đoán **lớp số 0** ở mức tương đối, nhưng đoán **lớp 4.0** ở mức tệ. Tỉ lệ xấp xỉ như khi ta đã thống kê ở tập train là mô hình có nhiều mẫu **lớp 0.0** và ít mẫu **lớp 4.0**.
* Đa số mô hình đoán nhầm lớp thứ 3 với các lớp khác. Như vậy để cải thiện mô hình ta cần tìm ra điểm đặc trưng mà gây nhiễu cho mô hình.

#### **4.1.2.2. Cân bằng mẫu**

##### **4.1.2.2.1. Sử dụng hàm**

|  |  |
| --- | --- |
|  |  |
|  |  |
| Hình a (tập train) | Hình b (tập test) |

###### Bảng 4.3. Kết quả mô hình khi sử dụng hàm để tăng mẫu tối thiểu

**Nhận xét:**

* Nhìn chung khi sử dụng hàm mô hình đã dự đoán lớp số 4 tốt hơn nhưng tổng quan các độ đo về lớp khác lại bị giảm xuống.
* Bị overfitting đối với lớp 4 ở tập train

##### **4.1.2.2.2. Sử dụng thư viện SMOTE**

|  |  |
| --- | --- |
|  |  |

###### Hình 4.5. Kết quả của mô hình khi sử thư viện smote để cân bằng mẫu

**Nhận xét:**

* Khi sử dụng Smote thì mô hình cũng tăng độ chính xác khi dự đoán lớp 4, nhưng cũng giảm các độ các lớp khác cũng bị giảm xuống
* Sự khác biệt khi dùng thư viện Smote mô hình đoán bị nhầm nhiều sang lớp 4

#### **4.1.2.3. Vấn đề 1 và Vấn đề 3.**

**Nhận xét:**

Do mẫu có các giá trị group\_id với số lượng xuất hiện không đều, bị lệch về phía bên trái nên không hiệu quả với phương pháp IQR. Ta sẽ xét sang số nhóm mà 1 người dùng vào và áp dụng IQR lên đó. Như vậy ta sẽ xem xét luôn vấn đề 3 ở mục này:

* Chỉ loại các giá trị ngoại lai

|  |  |
| --- | --- |
|  |  |

###### Hình 4.6. Kết quả của mô hình sau loại bỏ giá trị ngoại lai

* **Nhận xét**: Nhìn chung sau khi loại bỏ các giá trị ngoại lai từ các số lượng nhóm mà 1 người dùng tham gia thì mô hình có xu hướng đoán không thay mà còn bị giảm một xíu so với khi chưa áp dụng.

|  |  |
| --- | --- |
|  |  |

###### Hình 4.7. Kết quả của mô hình khi sử dụng số lượng nhóm tham gia của người dùng

* **Nhận xét**:
* Khi ta sử dụng thêm đặc trưng thì lớp 1 và lớp 3 mô hình đoán đúng nhỉnh hơn, lớp 1 giảm xuống 1 chút. Tổng quát thì độ chính xác và độ chấp nhận của mô hình đều tăng.
* Như vậy ta sẽ thêm đặc trưng độ dài của các nhóm làm thêm 1 đặc trưng mới

#### **4.1.2.4. Vấn đề 2**

|  |  |
| --- | --- |
|  |  |

###### Hinh 4.8. Kết quả của mô hình khi sử dụng mô hình Logistic để loại bỏ điểm ngoại lệ

**Nhận xé**t:

* Ở lớp 2, 3, 4 sau khi mô hình loại bỏ điểm ngoại lệ đều tăng lên, góp phần gia tăng độ chính xác cũng như độ chấp nhận được của mô hình.
* Nhưng ta cần xét đến vấn đề overfitting của mô hình sau khi đã loại bỏ điểm ngoại lệ.

|  |  |
| --- | --- |
|  |  |

###### Hình 4.9. Kết quả của mô hình khi kiểm tra lại trên tập train

**Nhận xét**:

Sau khi loại bỏ những dữ liệu nhiễu thì mô hình có dấu hiệu bị overfit khi xét ở độ đo accuracy. Nhưng đối với độ chấp nhận được thì chênh lệch 4% ta vẫn chấp nhận được.

#### **4.1.2.6. Vấn đề 4**

|  |  |
| --- | --- |
|  |  |

###### Hình 4.10. Kết quả của mô hình khi loại bỏ đi phần tử trùng nhau giữa 2 lớp kề nhau với số lượng nhóm tốt nhất là 15.

Nhận xét:

* Với việc thay đổi đặc trưng với điều kiện không quá lớn thì ta thấy các độ đo cũng thay đổi. Tổng thể các độ đo không giảm nhiều, lớp thứ 4 tăng ít.
* Nếu ta tăng số lượng nhóm lên thì các độ đo sẽ giảm, vì ta vô tình làm mất đặc trưng có được từ những nhóm bị loại, mà các nhóm còn lại không thể giúp mô hình học tổng quát được các đặc trưng
* Hơn nữa việc loại bỏ và tạo ra đặc trưng mới của vấn đề này rất rộng.

#### **4.1.2.7. Vấn đề 5**

|  |  |
| --- | --- |
|  |  |

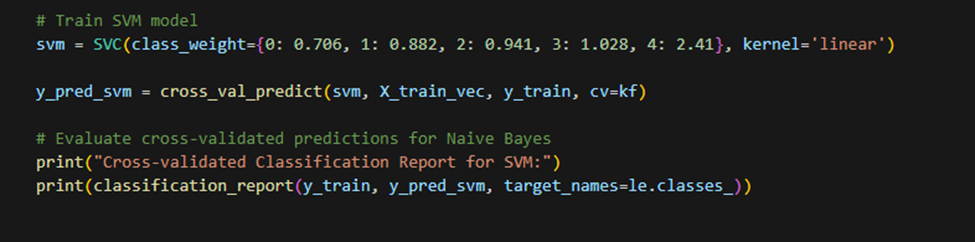
###### Hình 4.11. Kết quả của mô hình sau khi thêm đặc trưng từ độ dài id của nhóm

**Nhận xét**:

* Với việc thêm 6 cột độ dài từ id của nhóm thì nhìn chung các độ không thay đổi gì nhiều lắm. Độ chính xác lên 52 %, các độ đo khác không biến động gì nhiều. Chứng tỏ đặc đặc trưng hiện tại chưa ảnh hưởng đến việc mô hình phân lớp

## **4.2. Mô hình SVM (Support Vector Machine)**

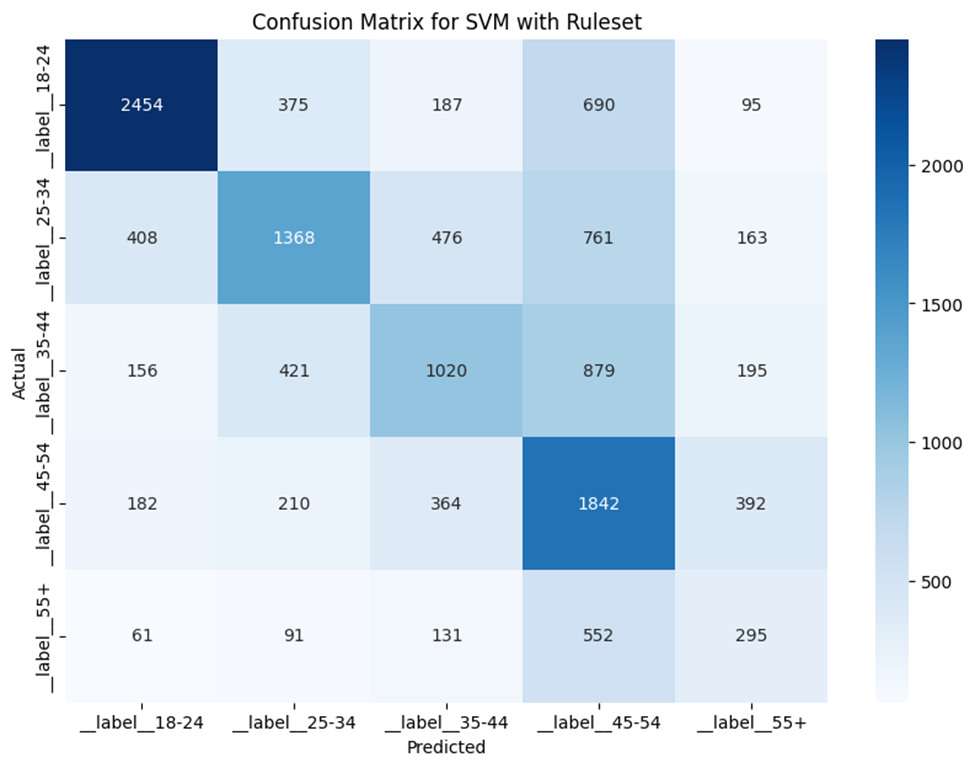
### **4.2.1. Khi sử dụng đặc trưng trích xuất từ thư viện**

Ta cũng sẽ sử dụng **classWeight** và **K-fold cross validation** tương tự Logistic Regression (Tham khảo 4.1.1.1)

###### Hình 4.12. Đoạn mã sử dụng classWeight và k-fold cho mô hình

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  |  | |
| Tập train Tập test |

###### Hình 4.13. Kết quả của mô hình khi sử dụng cross validation với k = 5 và class weight trên tập train và tập test



###### Hình 4.14. Ma trận nhầm lẫn tương ứng

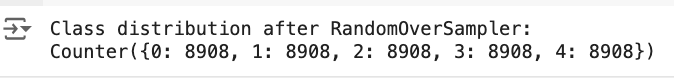
**Nhận xét:**

* Mô hình có tỉ lệ chính xác (precision) và recall tương đối cao ở nhóm tuổi 18-24 (0.73 và 0.66).
* Tỉ lệ precision và recall trung bình ở nhóm tuổi 25-34 và 35-44 (0.48-0.55 cho precision, 0.39-0.44 cho recall).
* Tỉ lệ recall tạm ổn cho nhóm tuổi 45-54 (0.66).
* Tỉ lệ precision và recall thấp ở nhóm tuổi 55+ (0.38 và 0.16).
* Mô hình vẫn có khả năng đoán gần đúng nhãn tuổi khá tốt (0.80).
* Mô hình thường xuyên bị nhầm nhãn tuổi 45-54 với 4 nhóm tuổi còn lại.

=> Chúng ta vẫn thấy được những vấn đề tương tự như khi sử dụng Logistic Regression, thậm chí còn có phần tệ hơn một chút. Mô hình đoán ổn ở 3 nhãn đầu, bị nhiễu và đoán sai nhiều ở nhãn thứ 4, và đoán cũng chưa tốt lắm ở nhãn thứ 5. Có lẽ sẽ cần phải tinh chỉnh lại để đưa ra kết quả tốt hơn.

## **4.3. Mô hình Random Forest**

**4.3.1. Khi sử dụng đặc trưng trích xuất từ thư viện**

* Sử dụng **RandomOversampler** để cân bằng tập dữ liệu.

### 

### 

### 

### 

### 

###### Tập train Tập test

**Nhận xét:** Tỷ lệ recall ở các nhóm 35-44, 45-54, 55+ khá thấp. Các chỉ số macro avg và weighted avg cũng không cao, cho thấy hiệu suất của mô hình trên các lớp khác nhau không đồng đều.

## **4.4. Mô hình LightGBM:**

Khi sử dụng các đặc trưng tối ưu dùng ở mô hình Logistic ta được:

|  |  |
| --- | --- |
|  |  |

###### Hình 4.15. Kết quả khi sử dụng mô hình Light gbm

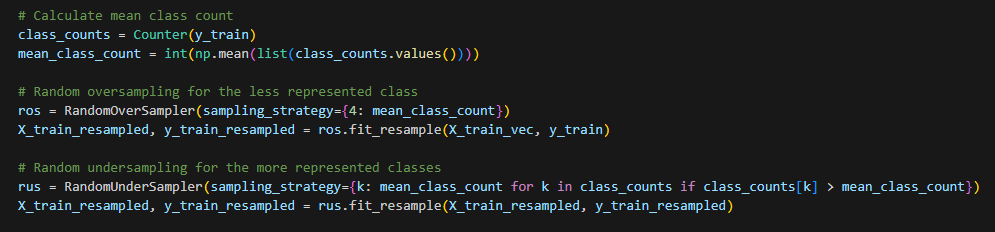
**Nhận xét:**

Ở mô hình này, lớp 1 được cải thiện nhưng lớp 2 và lớp 4 thụt lùi. Nhìn chung các độ đo không biến động quá nhiều. Và mô hình này với đặc trưng này chưa hiệu quả bằng Logistic Regression

## **4.5.Mô hình Naïve Bayes**

### **4.5.1. Khi sử dụng đặc trưng trích xuất từ thư viện**

Do mô hình Naive Bayes không hỗ trợ trọng số cho các lớp, nên thay vào đó ta sẽ sử dụng **RandomOversampler** và **RandomUndersampler** để cân bằng tập dữ liệu, và sử dụng **K-fold cross validation** (Tham khảo 4.1.1) để kiểm tra.



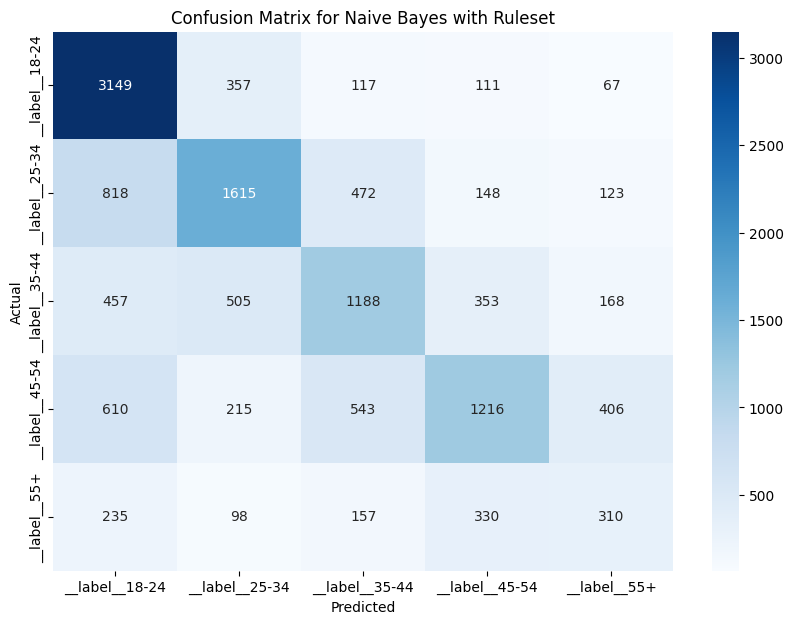
###### Hình 4.16. Sử dụng RandomOverSampler và RandomUnderSampler cho tập dữ liệu

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  |  | |
| Trước khi resample Sau khi resample |

###### Hình 4.17. Tập dữ liệu trước và sau khi chỉnh sửa

|  |
| --- |
| Hình 4.18. Đoạn mã sử dụng tập dữ liệu mới và k-fold cho mô hình |
| |  |  | | --- | --- | |  |  | |
| Tập train Tập test |

###### Hình 4.19. Kết quả của mô hình khi sử dụng cross validation với k = 5



###### Hình 4.20. Ma trận nhầm lẫn tương ứng

**Nhận xét:**

* Tỉ lệ recall ở nhóm tuổi 18-24 tốt (0.87).
* Tỉ lệ recall ở nhóm tuổi 55+ cực thấp (0.08).
* Chỉ số trung bình ở các nhóm còn lại.
* Khả năng đoán gần đúng tốt (0.82).

=> Ở đây, ta có thể thấy kết quả dự đoán của mô hình này đã khác hai mô hình trước. Naïve Bayes lại có xu hướng đoán nhầm nhóm tuổi 18-24 sang các nhóm tuổi khác hơn. Nhưng mà đã không còn tình trạng đoán nhầm ở nhóm 45-54 và độ chính xác chấp nhận được của nhóm 55+ cũng đã được cải thiện. Có thể thấy mô hình này có kết quả khả quan nhất trong 3 mô hình.

#### 

# **PHẦN 5: TỔNG KẾT**

## **5.1. Mô hình tốt nhất**

Dựa vào kết quả mà các mô hình thể hiện ở phần 4, ta có thể thấy rằng Logistic Regression đã đưa ra kết quả tốt nhất với các đặc trưng mà ta đã tìm được nếu như ta xét theo 2 góc độ là độ chính xác và độ chấp nhận được(kết quả lần lượt là 55% với độ chính xác, và độ chấp nhận được là 83%). Ngoài ra, nếu ta xét theo ma trận nhầm lẫn, Logistic Regression cũng có một phân bố kết quả trên đường chéo chính tốt hơn 2 mô hình còn lại. Vì vậy nên nó sẽ là mô hình tốt nhất cho bài toán này.

## **5.2. Các vấn đề còn tồn đọng**

* **Cân bằng dữ liệu**: Các phương pháp cân bằng dữ liệu hiện tại chưa hiệu quả.
* **Tạo đặc trưng**: Các cách xử lý để tạo thêm đặc trưng chưa hiệu quả dẫn đến các đặc trưng hiện tại vẫn chưa đủ tốt để mô hình có thể học tổng quát dữ liệu. Việc lựa chọn và tạo các đặc trưng mới cần phải được cải thiện để tăng cường khả năng của mô hình trong việc nhận diện các mẫu phân loại.
* **Nhầm lẫn nhóm tuổi**: Một số mô hình vẫn dễ bị nhầm lẫn nhóm tuổi 45-54 với các nhóm tuổi khác. Đây là vấn đề nghiêm trọng vì nhóm tuổi này có sự phân bố mẫu không đồng đều và có thể bị ảnh hưởng bởi các yếu tố không được đại diện rõ ràng trong dữ liệu hiện tại.

## **5.3. Hướng cải thiện**

* **Tạo đặc trưng**: Tiến hành phân tích sâu hơn để phát hiện và tạo ra các đặc trưng mới có thể cải thiện khả năng phân loại của mô hình. Các phương pháp như feature engineering nâng cao, kết hợp các đặc trưng hiện tại, và áp dụng các kỹ thuật lựa chọn đặc trưng tiên tiến nên được xem xét.
* **Cải thiện mô hình**:. Điều chỉnh các tham số của mô hình để tối ưu hóa hiệu suất, đặc biệt là trong việc phân loại các nhóm tuổi dễ bị nhầm lẫn.
* **Đánh giá và tinh chỉnh**: Tiến hành đánh giá chi tiết hơn các phương pháp hiện tại và tiến hành tinh chỉnh dựa trên các chỉ số hiệu suất. Thực hiện các bước thử nghiệm và cải tiến liên tục để đạt được kết quả tốt nhất.

# **PHẦN 6: PHÂN CHIA CÔNG VIỆC**

|  |  |  |
| --- | --- | --- |
| Họ và Tên | Công việc | Phần trăm |
| Nguyễn Hữu Cứ | Phân tích bài toán, xác định bài toán, Hiểu dữ liệu, Chuẩn bị dữ liệu, Mô hình Light Gbm, Viết báo cáo. | 20 |
| Võ Quang Sáng | Phân tích bài toán, xác định bài toán, Hiểu dữ liệu, Chuẩn bị dữ liệu, tìm hiểu về mô hình Logistic Regression, Làm slide . | 20 |
| Nguyễn Hoàng Điệp | Phân tích bài toán, xác định bài toán, Hiểu dữ liệu, Chuẩn bị dữ liệu, tìm hiểu về mô hình Random Forest, Viết báo cáo. | 20 |
| Lê Công Hoàng | Phân tích bài toán, xác định bài toán, Hiểu dữ liệu, Chuẩn bị dữ liệu, tìm hiểu về mô hình Naive Bayes, Viết báo cáo | 20 |
| Trương Đức Quang | Phân tích bài toán, xác định bài toán, Hiểu dữ liệu, Chuẩn bị dữ liệu, tìm hiểu về mô hình SVM, làm slide | 20 |