HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

KHOA VIỄN THÔNG I

BÁO CÁO

THỰC TẬP TỐT NGHIỆP

Đơn vị: Công ty Cổ phần Viễn thông FPT – FPT TELECOM

Giảng viên hướng dẫn	**:**	**Phạm Thị Thuý Hiền**
Sinh viên thực tập	**:**	**Trần Duy Lăng**
Mã sinh viên	**:**	**B20DCVT224**
Lớp	**:**	**D20VTHI03**
Vị trí thực tập	**:**	**Data Science**

Hà Nội, 2024

LỜI MỞ ĐẦU

Thực hiện nhiệm vụ học tập của nhà trường trong đợt thực tập tốt nghiệp, được sự đồng ý của ban lãnh đạo Khoa Viễn thông I và công ty FPT Telecom Bám sát đề cương của Khoa đưa ra, những lý luận và kiến thức đã học. Trong quá trình thực tập em đã được trau dồi thêm kiến thức về chuyên môn cũng như cơ cấu và tình hình hoạt động của công ty.

Được sự tận tình giúp đỡ hướng dẫn của các anh chị trong phòng Kỹ thuật hệ thống cùng với sự giúp đỡ của giáo viên hướng dẫn Phạm Thị Thu Hiền và nỗ lực trong học tập, tìm tòi học hỏi, em đã hoàn thành báo cáo thực tập tốt nghiệp của mình.

Báo cáo này gồm 2 phần chính:

- ***Phần I: Tổng quan đơn vị thực tập***
- ***Phần II: Báo cáo chuyên sâu***

Trong quá trình thực hiện báo cáo, tuy đã cố gắng nhưng em vẫn còn những hạn chế về thời gian tìm hiểu, kiến thức cũng như là kinh nghiệm và vẫn còn nhiều sai sót. Em rất mong được nhận những ý kiến đóng góp và nhận xét để em có thể hoàn thiện hơn.

LỜI CẢM ƠN

Để hoàn thành được báo cáo thực tập tốt nghiệp này thì em xin cảm ơn tới phía ban lãnh đạo Công ty Cổ phần FPT Telecom đã tạo điều kiện cho em được thực tập tại đây. Ngoài ra, em cảm ơn anh Dương Công Hậu đã luôn nhiệt tình chỉ dẫn, giảng dạy cho em những kiến thức về khoa học dữ liệu để em có thể hoàn thiện hơn.

Em xin gửi lời cảm ơn tới ban lãnh đạo Học viện, Khoa đào tạo đã thiết lập khung chương trình có môn học "Thực tập" để em có thể vận dụng những kiến thức học được trên giảng đường áp dụng vào công việc và được tham quan, tìm hiểu về môi trường doanh nghiệp.

Em xin gửi lời cảm ơn sâu sắc và chân thành tới giảng viên hướng dẫn Phạm Thị Thuý Hiền. Nhờ cô luôn dìu dắt, giúp đỡ và chỉ bảo tận tình cho em để hoàn thành tốt Thực tập.

Em xin chân thành cảm ơn!

Hà Nội, ngày 15 tháng 08 năm 2024

MỤC LỤC

DANH MỤC HÌNH ẢNH

PHẦN I: TỔNG QUAN VỀ ĐƠN VỊ THỰC TẬP

1.1 Giới thiệu về đơn vị thực tập

Địa chỉ: số 48 Vạn Bảo, phường Ngọc Khánh, quận Ba Đình, thành phố Hà Nội

Hình 1 Hình ảnh của công ty

1.2 Lịch sử hình thành và phát triển

- 31/1/1997: Thành lập Trung tâm Dữ liệu trực tuyến FPT (FPT Online Exchange – FOX)
- 2001: Ra mắt trang báo điện tử đầu tiên tại Việt Nam
- 2002: Trở thành nhà cung cấp kết nối Internet IXP (Internet Exchange Provider)
- 2005: Chuyển đổi thành Công ty Cổ phần Viễn thông fpt (FPT Telecom)
- 2007: FPT Telecom bắt đầu mở rộng hoạt động trên phạm vi toàn quốc, được cấp giấy phép cung cấp dịch vụ viễn thông liên tỉnh và cổng kết nối quốc tế. Đặc biệt, FPT TEL đã được trở thành thành viên chính thức của Liên minh AAG.
- 2008: Trở thành nhà cung cấp dịch vụ Internet cáp quang băng rộng (FTTH) đầu tiên tại Việt Nam và chính thức có đường kết nối quốc tế từ Việt Nam đi Hồng Kông.

- 2009: Đạt mốc doanh thu 100 triệu đô la Mỹ và mở rộng thị trường sang các nước lân cận như Campuchia.
- 2012: Hoàn thiện tuyến trục Bắc – Nam với tổng chiều dài 4000km đi qua 30 tỉnh thành.
- 2014: Tham gia cung cấp dịch vụ truyền hình IPTV với thương hiệu Truyền hình FPT.
- 2015: FPT Telecom có mặt trên cả nước với gần 200 VPGD, chính thức được cấp phép kinh doanh tại Myanmar, đạt doanh thu hơn 5,500 tỷ đồng và là một trong những đơn vị dẫn đầu trong triển khai chuyển đổi giao thức liên mạng IPv6.
- 2016: Khai trương Trung tâm Dữ liệu FPT Telecom mở rộng chuẩn Uptime TIER III với quy mô lớn nhất miền Nam. Được cấp phép triển khai thử nghiệm mạng 4G tại Việt Nam. Đồng thời là doanh nghiệp Việt Nam đầu tiên nhận giải thường Digital Transformers of the Year của IDC.

1.3 Nhiệm vụ

Sứ mệnh của FPT là tiên phong đưa Internet đến với người dân Việt Nam và mong muốn mỗi gia đình Việt Nam đều sử dụng ít nhất một dịch vụ của FPT Telecom, đồng hành cùng phương châm " Khách hàng là trọng tâm", chúng tôi không ngừng nỗ lực đầu tư hạ tầng, nâng cấp chất lượng sản phẩm – dịch vụ, tăng cường ứng dụng công nghệ mới để mang đến cho khách hàng những trải nghiệm sản phẩm dịch vụ vượt trội.

PHẦN II: BÁO CÁO CHUYÊN SÂU

1. Đề tài tham gia

1.1 Nguyên nhân và mục đích nghiên cứu

Trong quá trình thực tập tại FPT Telecom, em được tham gia vào một dự án liên quan đến việc xử lý dữ liệu thực tế từ các trạm pop. Theo đó, tại có các trạm pop có hệ thống IoT, có nhiệm vụ đo những **thông số trạng thái của** trạm và gửi về platform được xây dựng trước của công ty.

Thông thường, mỗi trạm POP sẽ chứa một hoặc một vài bộ OLT (trong cấu trúc FTTx), cung cấp dịch vụ cho một khu vực lớn. Do đó, một vấn đề lớn được đặt ra là việc phải duy trì được trạng thai hoạt động của trạm POP 24/7, cho dù bất kỳ hoàn cảnh nào. Trong những điều kiện bình thường, các trạm POP sẽ sử dụng trực tiếp nguồn điện từ điện lưới quốc gia, và trong mỗi trạm POP sẽ luôn có thêm một bộ ac-quy cung cấp điện trong những trường hợp mất điện. Bình này sẽ được nạp điện đầy trở lại khi sự cố mất điện kết thúc.

Vấn đề được đặt ra đó là thời gian hoạt động của ac-quy có thể đảm bảo duy trì hoạt động cho trạm đến khi có điện trở lại hay không? Trung bình điện áp phục vụ trong một trạm POP thường trên 43V. Một bình ac-quy sẽ phải cung cấp một điện áp lớn hơn 43V cho trạm POP, nếu không, trạm POP sẽ không thể hoạt động, gây ảnh hưởng lớn đến việc cung cấp dịch vụ của FPT cho địa phương đó. Để tránh việc ac-quy chạy đến kiệt, các đơn vị sẽ phân công nhân viên mang máy phát điện đến để "cứu" những trạm POP này. Nhưng tại mỗi đơn vị chỉ có số lượng máy phát nhất định, do đó cần phải có cách để phân công một cách hợp lý để đảm bảo "cứu" đúng trạm, đúng thời điểm.

Phương pháp được đề ra đó là dự đoán **trong trường hợp mất điện lâu dài, xác định thời gian ac-quy còn có thể cấp điện cho trạm POP**. Theo đó, chúng ta sẽ cần dữ liệu thông tin các trạm POP trong môt sự cố mất điện, sàng lọc và trích xuất dữ liệu cũng như huấn luyện mô hình học máy để có thể dự đoán thời gian hoạt động của ac-quy. Và từ thời gian được dự đoán, các đơn vị có thể phân công máy phát đến những trạm POP đang cần gấp, đảm bảo có thể duy trì hoạt động cho trạm POP, không làm gián đoạn dịch vụ đến khách hàng.

1.2 Dữ liệu cần thiết

Để có thể huấn luyện được mô hình học máy, chúng ta cần có dữ liệu thích hợp. Với mục tiêu xác định được thời gian bình ac-quy có thể hoạt động trong lần mất điện tiếp theo, chúng ta cần phải có dữ liệu cho những lần mất điện trong quá khứ. Tại mỗi trạm POP đều được xây dựng một hệ thống IoT, cung cấp dữ liệu (với tần suất 2 phút một lần) về trạng thái của ac-quy, ví dụ như nhiệt độ, điện áp, tải…

```
RangeIndex: 248458 entries, 0 to 248457
Data columns (total 19 columns):
 #   Column                 Non-Null Count   Dtype
---  ------                 --------------   -----
 0   _created_at            248458 non-null  object
 1   query_rs               248458 non-null  object
 2   ip                     248458 non-null  object
 3   host_name              248458 non-null  object
 4   pop_name               248458 non-null  object
 5   province               248458 non-null  object
 6   department             248458 non-null  object
 7   business_zone          248458 non-null  object
 8   brand                  248458 non-null  object
 9   service_lost_output    248458 non-null  object
 10  power_lost_state       196100 non-null  object
 11  dc_volt_output         196156 non-null  float64
 12  ac_volt_input          192572 non-null  float64
 13  load_curr              196146 non-null  float64
 14  batt_temp              196013 non-null  float64
 15  curr_power_lost_start  194218 non-null  object
 16  curr_power_lost_end    0 non-null       float64
 17  prev_power_lost_start  85964 non-null   object
 18  prev_power_lost_end    86993 non-null   object
dtypes: float64(5), object(14)
memory usage: 36.0+ MB
```

Hình 2 Các cột trong dữ liệu ban đầu

Dữ liệu thu thập được là dữ liệu về sự kiện mất điện của các trạm POP của công ty. Dữ liệu này có chứa một số thông tin không được phép cung cấp ra bên ngoài. Trong bài báo cáo này, những thông tin này sẽ không được phân tích.

```
RangeIndex: 248458 entries, 0 to 248457
Data columns (total 14 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   _created_at           248458 non-null  object
 1   query_rs              248458 non-null  object
 2   pop_name              248458 non-null  object
 3   brand                 248458 non-null  object
 4   service_lost_output   248458 non-null  object
 5   power_lost_state      196100 non-null  object
 6   curr_power_lost_start 194218 non-null  object
 7   curr_power_lost_end   0 non-null       float64
 8   prev_power_lost_start 85964 non-null   object
 9   prev_power_lost_end   86993 non-null   object
 10  dc_volt_output        248458 non-null  object
 11  ac_volt_input         248458 non-null  object
 12  batt_temp             248458 non-null  object
 13  load_curr             248458 non-null  object
dtypes: float64(1), object(13)
memory usage: 26.5+ MB
```

Hình 3 Thông tin về các trường dữ liệu ban đầu

Trong nhưng trường dữ liệu này, "query_rs" là một json chứa những dữ liệu từ các máy đo và cảm biến được lắp trong hệ thống IoT, đo các chỉ số liên quan của bình ac-quy có trong POP. Chúng ta có thể trích xuất ra thông tin về tải, nhiệt độ, nhiệt độ bình,… một cách chính xác và đầy đủ hơn.

'{"ac_alarm": 0, "relay_ac": "NOSUCHOBJECT", "sys_name": "Enatel SM3X NT2", "batt_curr": 2.6, "batt_temp": 34.3, "batt_test": "NOSUCHOBJECT", "load_curr": 3.7, "rect_curr": 6.3, "sys_descr": "Enatel SM3X NT2 SNMP Agent", "sys_uptime": 99237164, "version_fw": 118, "alarm_relay": 24, "batt_remain": 100.0, "batt_status": 0, "rect_number": 2, "ac_phase_lost": "NOSUCHOBJECT", "ac_volt_input": 221.0, "dc_volt_output": 53.33, "relay_generator": "NOSUCHOBJECT"}'

Hình 4 Ví dụ về một giá trị trong cột query_rs

```
import json
df1.drop(["dc_volt_output","ac_volt_input","batt_temp","load_curr"], axis = 1, inplace= True)
df1["query_rs"] = df1.apply(lambda row: json.loads(row['query_rs']),axis = 1)
df1['dc_volt_output'] = df1['query_rs'].apply(lambda x: x['dc_volt_output'])
df1['ac_volt_input'] = df1['query_rs'].apply(lambda x: x['ac_volt_input'])
df1['batt_temp'] = df1['query_rs'].apply(lambda x: x['batt_temp'])
df1['load_curr'] = df1['query_rs'].apply(lambda x: x['load_curr'])
```

Hình 5 Trích xuất dữ liệu từ giá trị trong cột query_rs

Lúc này đây, chúng ta đã có dữ liệu của những lần sự cố mất điện tại các POP. Đối với mỗi lần mất điện, trạm POP sẽ gửi về các bản tin với tần suất 2 phút1 bản tin. Mỗi một bản tin sẽ bao gồm các thông tin như trong bảng sau:

STT	Tên	Ý nghĩa
1	_created_at	Thời gian bản tin gửi đến
2	pop_name	Tên trạm POP

3	service_lost_output	Cho biết POP mất điện đang chạy bình hay chạy máy phát
4	power_lost_state	Trạng thái tủ POP
5	prev_power_lost_start	Thời gian bắt đầu của sự cố mất điện trước
6	prev_power_lost_end	Thời gian kết thúc của sự cố mất điện trước
7	dc_volt_output	Điện áp của bình ac-quy tại thời điểm đó
8	batt_temp	Nhiệt độ bình tại thời điểm đó
9	load_curr	Tải của bình tại thời điểm đó

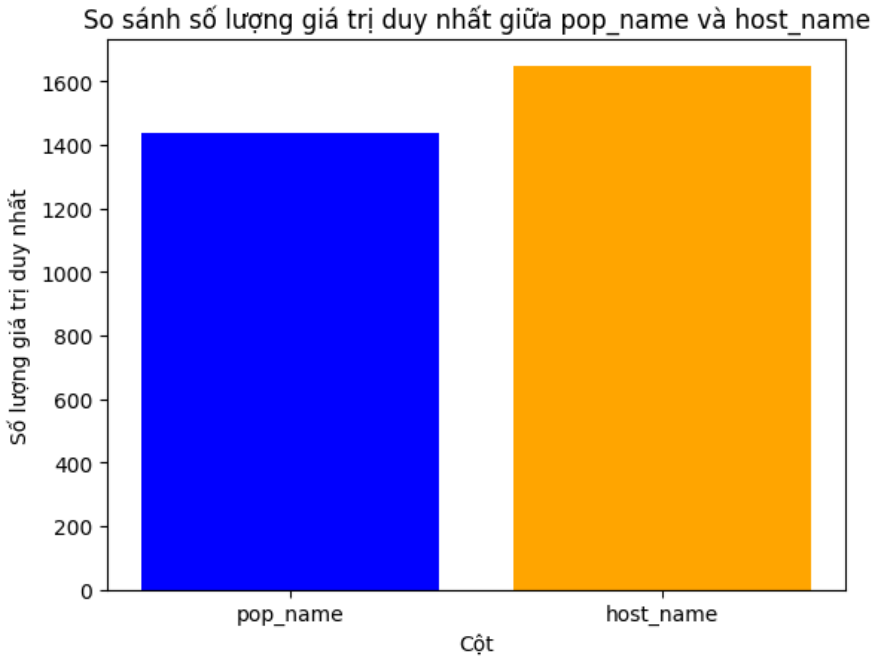
Tuy nhiên, dữ liệu trên đây mới chỉ là dữ liệu thô, chúng ta không thể sử dụng dữ liệu này cho việc phân tích và huấn luyện mô hình Máy học. Chúng ta cần xử lý thêm để có thể trích xuất thành một bảng dữ liệu mới, thích hợp hơn cho việc phân tích và huấn luyện mô hình.

2. Xử lý dữ liệu thô

Mục tiêu của việc xử lý dữ liệu thô đó là tạo thành một dữ liệu mới có thể sử dụng được cho việc phân tích và xây dựng mô hình học máy. Theo đó, dữ liệu thô được thu thập từ các trạm POP sẽ được xử lý sao cho trở thành tập dữ liệu có ý nghĩa. Do mục tiêu của chúng ta là dự đoán thời gian bình ac-quy có thể cấp điện cho trạm POP trong trường hợp mất điện lâu dài, chúng ta cần có thông tin về những sự kiện mất điện đã xảy ra trong quá khứ.

2.1 Tách các sự kiện mất điện riêng biệt

Nhiệm vụ đầu tiên của chúng ta sẽ là tách tập dữ liệu thành những sự kiện mất điện riêng lẻ. Nhìn chung, để có thể tách các sự kiện mất điện, các bản tin trong cùng một sự kiện mất điện phải có một điểm chung nào đó với nhau. Trước tiên, số lượng pop_name và host_name đang không bằng nhau, chúng ta có thể thấy thông qua hình dưới đây:



Hình 6 So sánh số lượng trong pop_name và host_name

Thông thường, mỗi pop_name chỉ có tương ứng với một host_name. Việc số host_name cao hơn số pop_name cho thấy tồn tại một số trạm POP có nhiều hơn một host. Điều này là không hợp lý và cần phải loại bỏ những pop_name này. Sử dụng hàm **groupby()** nhóm theo pop_name và lọc những pop có nhiều hơn 1 host_name, sau đó loại bỏ những host_name này ra khỏi tập dữ liệu.

```
<class 'pandas.core.frame.DataFrame'>
Index: 213549 entries, 0 to 248457
Data columns (total 15 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   _created_at           213549 non-null  object
 1   query_rs              213549 non-null  object
 2   host_name             213549 non-null  object
 3   pop_name              213549 non-null  object
 4   brand                 213549 non-null  object
 5   service_lost_output   213549 non-null  object
 6   power_lost_state      168994 non-null  object
 7   curr_power_lost_start 167249 non-null  object
 8   curr_power_lost_end   0 non-null       float64
 9   prev_power_lost_start 72226 non-null   object
 10  prev_power_lost_end   73171 non-null   object
 11  dc_volt_output        213549 non-null  object
 12  ac_volt_input         213549 non-null  object
 13  batt_temp             213549 non-null  object
 14  load_curr             213549 non-null  object
dtypes: float64(1), object(14)
memory usage: 26.1+ MB
```

Hình 7 Thông tin dữ liệu sau khi lọc những pop không thích hợp

Như vậy, có khoảng 30 nghìn bản ghi đã bị loại bỏ sau khi lọc những pop có nhiều hơn 1 host. Để có thể tìm hiểu sâu hơn những vấn đề có trong tập dữ liệu, chúng ta hãy chọn ngẫu nhiên một vài trạm pop để xem quá trình mất điện của chúng. Trước hết, lấy dữ liệu trạm THAP033 và sắp xếp các bản ghi theo thời gian tạo của bản tin đó:

_created_at	pop_name	service_lost_output	power_lost_state	prev_power_lost_start	prev_power_lost_end
2024-05-30 15:50:01.947	THAP033	CRITICAL: CUP DIEN - CHAY BINH	BAT DAU MAT DIEN	NaN	NaN
2024-05-30 15:52:02.754	THAP033	CRITICAL: CUP DIEN - CHAY BINH	DANG MAT DIEN	NaN	NaN
2024-05-30 15:54:02.000	THAP033	CRITICAL: CUP DIEN - CHAY BINH	DANG MAT DIEN	NaN	NaN

Hình 8 Một số bản tin của trạm THAP033

Nhìn vào đây, chúng ta có thể thấy, bắt đầu mỗi sự kiện mất điện, trường power_lost_state sẽ có giá trị là "BAT DAU MAT DIEN", và trong suốt quá trình mất điện, trường này sẽ có giá trị là "DANG MAT DIEN". Đây là một dữ liệu quan trọng để có thể phân tách dữ liệu thành những sự kiện mất điện riêng biệt. Ngoài ra, đối với sự kiện mất điện đầu tiên của trạm (bắt đầu từ 15:50:01 30-05-2024), giá trị của các trường prev_power_lost_start và prev_power_lost_end (thời gian bắt đầu và kết thúc của lần mất điện trước đó) đang là NaN, chúng ta cũng cần điền những giá trị này.

Một trường hợp thường xuyên xảy ra nữa, đó là hai bản tin liên tiếp nhau các nhau thời gian trên 2 phút. Hệ thống IoT đang được thiết kế để có thể lấy và gửi thông tin cách nhau 2 phút. Việc hai bản tin cách nhau quá 2 phút có thể do 2 lý do:

- Hai bản tin nằm ở 2 sự cố mất điện khác nhau, thông thường, khoảng thời gian này sẽ rất lớn, khoảng vài tiếng đồng hồ. Điều này cũng có thể tận dụng để tách các sự kiện mất điện riêng biệt.
- Lỗi trong quá trình vận hành hệ thống IoT, có thể do đường truyền hoặc giao thức là mất bản tin ở giữa, khiến cho khoảng thời gian giữa hai bản tin liên tiếp lớn hơn 2 phút, nhưng cũng không quá lớn (thông thường chỉ mất một bản tin). Do đó, những trường hợp 2 bản tin cách nhau khoảng thời gian dưới 5 phút vẫn tính là trong cùng một sự cố mất điện.

2024-06-18 23:56:03.033	THAP033	WARNING: CUP DIEN - DANG CHAY MAY PHAT	DANG MAT DIEN	2024-06-15 16:10:03.562	2024-06-15 16:36:02.784
2024-06-19 00:00:02.929	THAP033	WARNING: CUP DIEN - DANG CHAY MAY PHAT	BAT DAU MAT DIEN	NaN	NaN

Hình 9 Ví dụ minh hoạ

Tuy nhiên, trong lý do thứ 2, có một trường hợp ít xảy ra, đó là hệ thống nhận bản tin đến sau là bản tin bắt đầu của sự cố mất điện mới (tức giá trị trường power_lost_state là "BAT DAU MAT DIEN"). Ví dụ như trong hình 9, hai bản tin đến vào lúc 2024-06-18 23:56:03.033 và 2024-06-29 00:00:02.929 rõ ràng là trong cùng một sự cố mất điện, những bản tin thứ hai lại bị nhận nhầm thành bản tin bắt đầu của sự kiện mới. Điều này làm ảnh hưởng đến việc xác định được những sự cố mất điện của trạm. Cần đặt lại giá trị của trường power_lost_state lại cho hợp lý để không phân tách sự kiện mất điện sai.

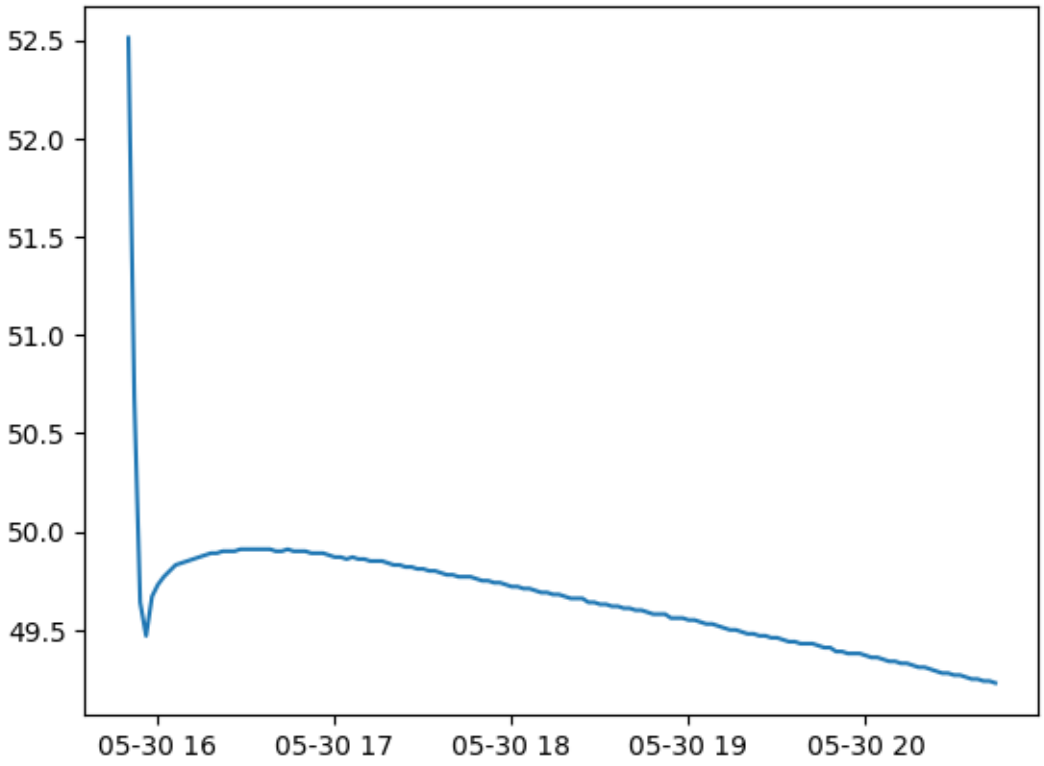
Sau khi xem xét một lượt, em đưa ra một cách để nhóm các bản tin trong một sự kiện mất điện riêng lẻ với nhau bằng cách nhóm các bản tin có giá trị prev_power_lost_start và prev_power_lost_start với nhau. Mỗi một sự kiện mất điện sẽ được xác định bằng thời gian bắt đầu và thời gian kết thúc của lần mất điện trước của nó. Điều này là khả thi hầu hết các bản tin đều chứa đầy đủ thông tin 2 trường này, ngoài trừ sự cố mất điện đầu tiên. Ngoài ra, vấn đề đặt sai giá trị trong trường "power_lost_state" cũng cần được xử lý.

Sau khi xử lý, chúng ta có được các sự kiện mất điện riêng lẻ có trong tập dữ liệu của trạm THAP033, bắt đầu từ ngày 30-05-2024 đến 01-07-2024.

```
Event 0: (Timestamp('2024-01-01 00:00:00'), Timestamp('2024-01-01 00:00:00'))
Event 1: (Timestamp('2024-05-30 15:50:01.947000'), Timestamp('2024-05-31 01:30:01.923000'))
Event 2: (Timestamp('2024-05-31 01:36:01.709000'), Timestamp('2024-05-31 02:28:02.454000'))
Event 3: (Timestamp('2024-06-06 22:34:03.064000'), Timestamp('2024-06-07 17:32:03.289000'))
Event 4: (Timestamp('2024-06-15 16:10:03.562000'), Timestamp('2024-06-15 16:34:03.507000'))
Event 5: (Timestamp('2024-06-18 14:56:03.062000'), Timestamp('2024-06-19 09:34:03.096000'))
Event 6: (Timestamp('2024-06-19 19:26:02.959000'), Timestamp('2024-06-19 19:36:03.655000'))
Event 7: (Timestamp('2024-06-22 17:48:02.709000'), Timestamp('2024-06-22 20:02:02.684000'))
```

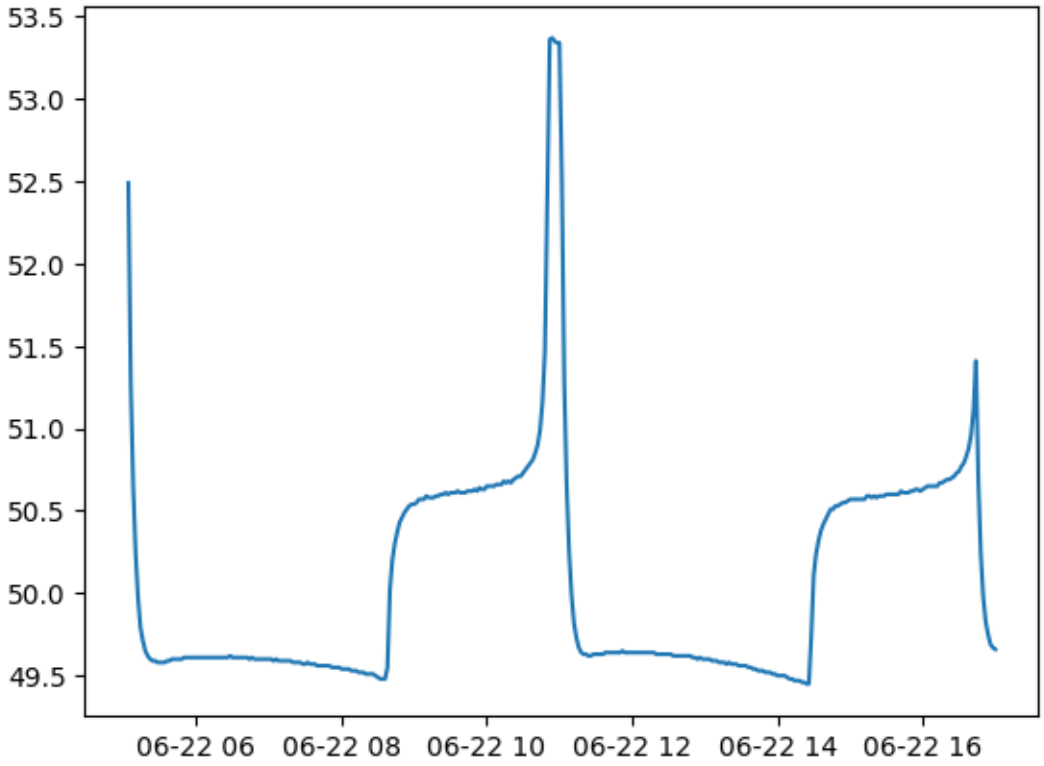
Hình 10 Các sự kiện mất điện của trạm THAP033 trong dữ liệu

Tuy nhiên, có một chú thích rằng, trong một sự kiện mất điện, có thể có những lần mất điện chạy bình và mất điện chạy máy phát được sử dụng xen kẽ nhau (thường xuất hiện trong những sự cố mất điện nhiều giờ). Và những lần chạy máy phát, chúng ta có thể coi là những lần có điện. Vì thể, một sự kiện mất điện này có thể có nhiều sự kiện mất điện con. Một lần mất điện chỉ chạy bình có sự thay đổi điện áp như sau:



Hình 11 Biểu diễn một lần mất điện chạy bình

Chúng ta có thể thấy, điện áp khi bắt đầu sử dụng bình sẽ giảm rất nhanh (do việc sử dụng bình để khởi động trạm POP mất rất nhiều năng lượng trong khoảng thời gian đầu). Sau đó, điện áp của bình sẽ giảm từ từ. Còn với một sự cố mất điện mà có cả chạy bình lẫn máy phát, điện áp sẽ có dạng như sau:



Hình 12 Biểu diễn một lần mất điện có nhiều lần chạy bình

Có thể dễ dàng nhận thấy những lần mất điện chạy bình có điện áp đi xuống và duy trì thấp, nhưng khi chạy máy phát điện, điện áp của bình sẽ nhanh chóng trở lại mức bình thường trước khi mất điện. Do đó, chúng ta sẽ coi những lần mất điện chạy bình trong mỗi sự cố mất điện sẽ là một sự cố mất điện mới.

Kết luận lại, để có thể nhóm được những sự cố mất điện từ dữ liệu, chúng ta cần loại bỏ những pop không hợp lệ. Sau đó cập nhật lại giá trị của trường power_lost_state cho hợp lý và điền đầy đủ thông tin 2 trường prev_power_lost_start và prev_power_lost_end. Sau đó sử dụng hàm **groupby()** nhóm theo thời gian bắt đầu và kết thúc của lần mất điện trước đó để tách thành những sự kiện mất điện. Cuối cùng, đối với những sự cố mất điện có nhiều lần phải chạy bình, chúng ta sẽ coi những lần chạy bình đó là một sự cố mất điện riêng biệt.

2.2 Trích xuất thông tin từ những sự cố mất điện

Mỗi sự cố mất điện sẽ cần có những đặc trưng riêng biệt. Từ trường _created_at, chúng ta có thể tính toán thời gian diễn ra của từng lần mất điện, và thời gian giữa những lần mất điện liên tiếp. Tuy nhiên, thời gian duy trì của bình trong lần mất điện tiếp theo phần lớn sẽ chịu ảnh hưởng của khoảng 2 đến 3 lần mất điện trước đó. Theo đó, đối với mỗi trạm POP, chúng ta sẽ lấy dữ liệu từ 2 lần mất điện trước đó để dự đoán ra nhãn cho lần mất điện tiếp theo. Để có thể tận dụng được tối đa dữ liệu, trong mỗi pop, chúng ta sẽ nhóm những sự cố mất

điện thành những bộ 3 sự cố, trong đó sự cố thứ 3 sẽ dùng để gán nhãn, hai sự cố đầu tiên sẽ sử dụng để trích xuất dữ liệu.

Chúng ta vẫn còn một vài trường dữ liệu chưa sử dụng như dc_volt_output, batt_temp, load_curr,…, những trường này đều là dữ liệu liên tục (numery). Đối với dc_volt_output, chúng ta sẽ tính toán tốc độ giảm điện áp trong mỗi lần xảy ra sự cố. Tương tự, chúng ta có được nhiệt độ trung bình của bình ac-quy và tải trung bình trong mỗi lần diễn ra sự cố mất điện.

Kết quả cuối cùng, chúng ta có được bảng dữ liệu mới có dạng như sau:

time_lost_1	time_lost_2	v_decrease_1	temp_1	load_1	v_decrease_2	temp_2	load_2	label
12.27	2.63	0.021750	32.73	2.51	0.005800	27.54	2.01	2_to_4_hours
19.10	2.93	0.001302	34.44	4.12	0.005395	33.52	4.14	2_to_4_hours
962.93	9.63	0.013558	29.68	3.25	0.010079	29.08	3.29	1_to_2_hours
61.17	573.83	0.024000	29.76	2.40	0.002237	28.58	2.40	2_to_4_hours
55.13	231.73	0.003583	29.69	1.89	0.016429	27.95	1.70	2_to_4_hours

Hình 13 Một số bản tin trong tập dữ liệu mới

Trong đó:

- time_lost_1 là khoảng thời gian giữa lần mất thừ nhất và thứ 2
- time_lost_2 là khoảng thời gian giữa lần mất thứ 2 và thứ 3 (tính theo giờ)
- v_decrease_1, temp_1, load_1 là giá trị tốc độ sụt áp, nhiệt độ trung bình và tải trung bình trong lần mất điện thứ nhất
- v_decrease_2, temp_2, load_2 là giá trị tốc độ sụt áp, nhiệt độ trung bình và tải trung bình trong lần mất điện thứ hai
- Label gồm các giá trị: less_1_hour, 1_to_2_hours, 2_to_4_hours, over_4_hours.

Chúng ta sẽ dùng dữ liệu này để xây dựng mô hình dự đoán thời gian bình ac-quy có thể chạy trong lần mất điện tiếp theo.

3. Xây dựng mô hình

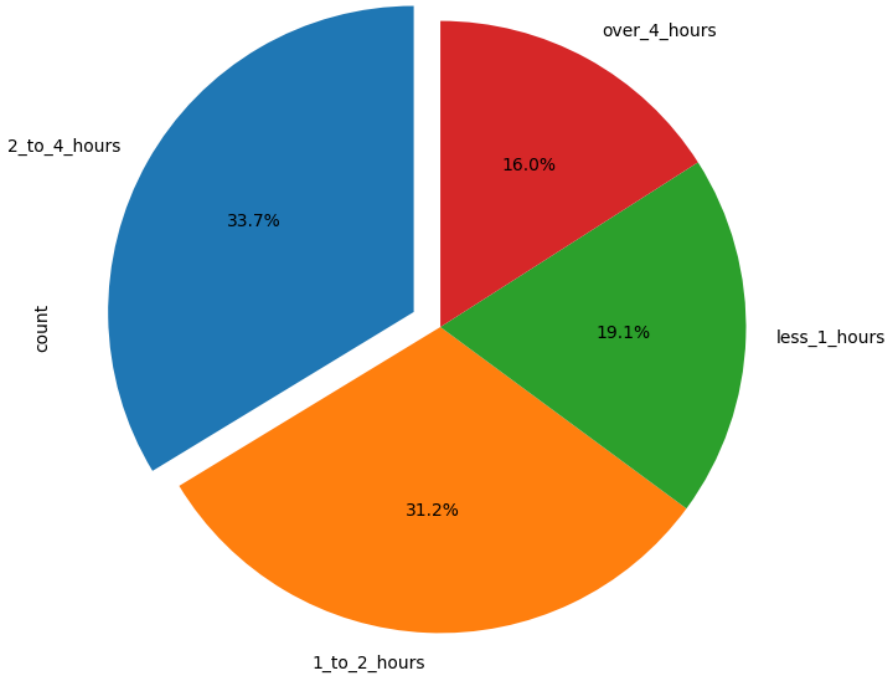
3.1 Những vấn đề có trong tập dữ liệu mới

Sau khi xử lý tập dữ liệu ban đầu, chúng ta đã có một tập dữ liệu mới với những sự khác biệt. Đầu tiên, hãy kiểm tra thông tin của tập dữ liệu này:

```
RangeIndex: 282 entries, 0 to 281
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   time_lost_1   282 non-null    float64
 1   time_lost_2   282 non-null    float64
 2   v_decrease_1  282 non-null    float64
 3   temp_1        282 non-null    float64
 4   load_1        282 non-null    float64
 5   v_decrease_2  282 non-null    float64
 6   temp_2        282 non-null    float64
 7   load_2        282 non-null    float64
 8   label         282 non-null    object
dtypes: float64(8), object(1)
memory usage: 20.0+ KB
```

Hình 14 Thông tin về các cột dữ liệu trong tập dữ liệu mới

Như đã đề cập ở trên, tập dữ liệu này bao gồm 8 trường dữ liệu, trong đó có một trường là nhãn để dự đoán. Tập dữ liệu mới này chỉ có khoảng 282 dòng, tương đối khiêm tốn cho việc xây dựng mô hình.

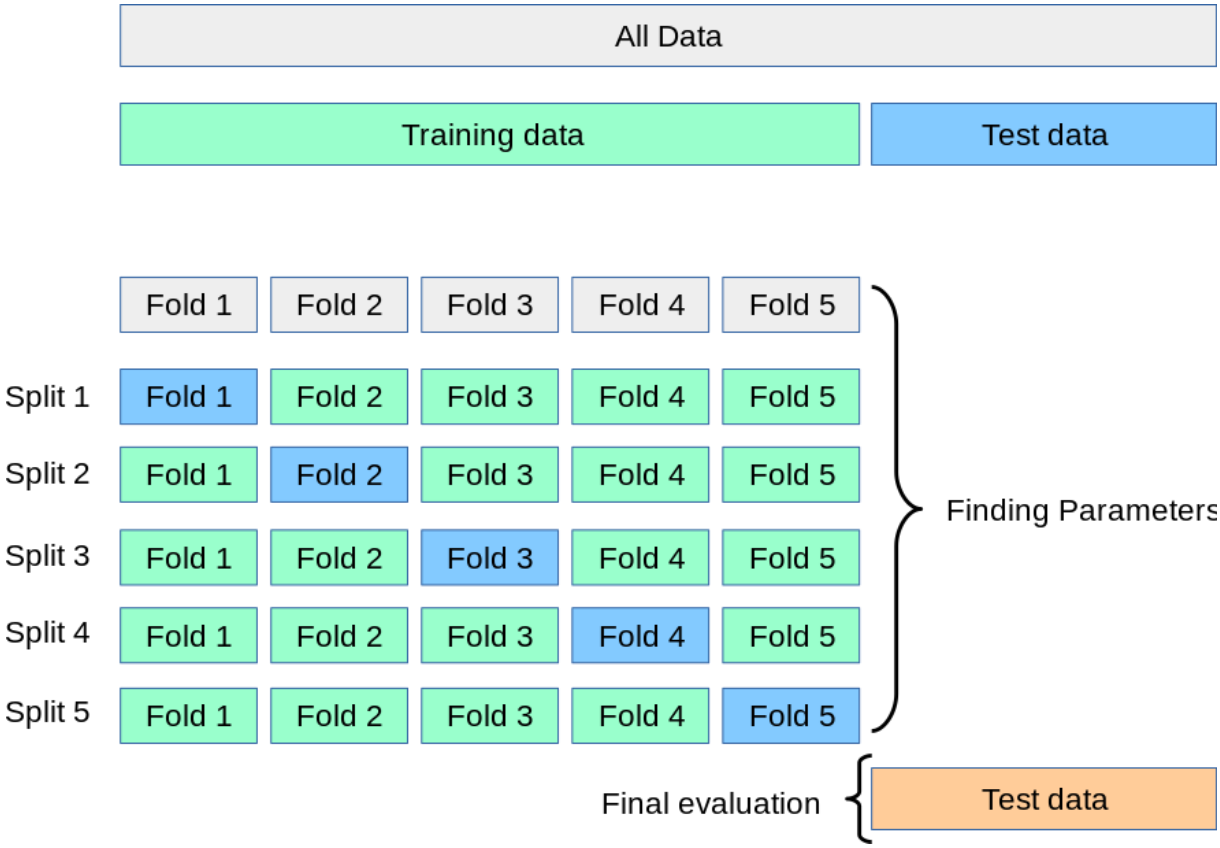


Hình 15 Phân bố các nhãn trong tập dữ liệu

Việc gán nhãn thích hợp giúp cho dữ liệu không bị mất cân bằng giữa các nhãn, nhờ đó mà chúng ta không phải xử lý việc mất cân bằng dữ liệu. Như vậy, vấn đề lớn nhất của tập dữ liệu này đó là việc nó tương đối khiêm tốn, điều này ảnh hưởng rất lớn đến hiệu quả của mô hình huấn luyện sau này.

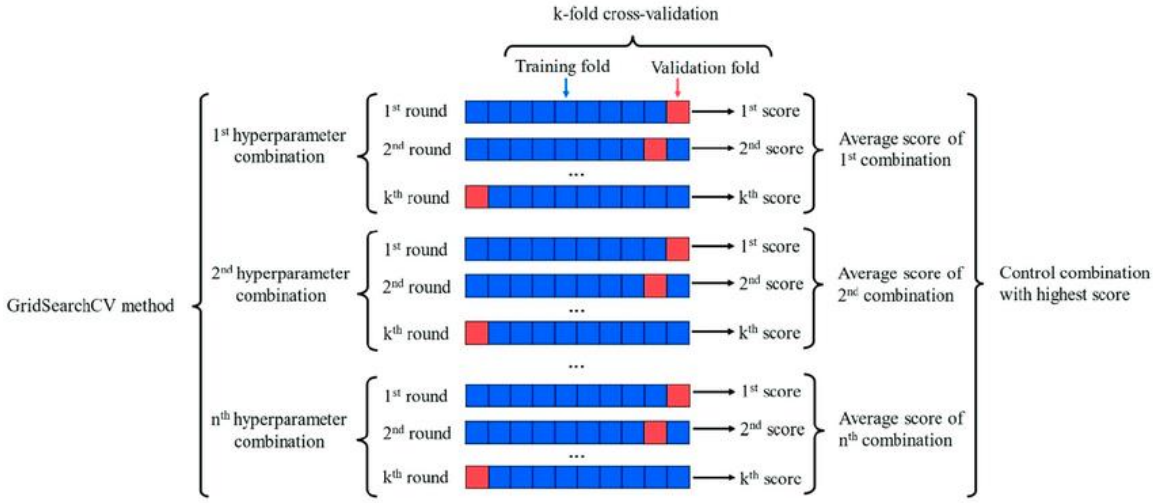
3.2 Xây dựng mô hình

Để giải quyết vấn đề thiếu dữ liệu, giải pháp tốt nhất đó là thu thập thêm dữ liệu. Tuy nhiên, đối với trường hợp dữ liệu khó thu thập, hoặc là yêu cầu phải xây dựng từ tập dữ liệu này, chúng ta cần có những giải pháp khác. Trong bài này, chúng ta sẽ sử dụng phương pháp PCA (Principal Component Analysis) để đơn giản hoá mô hình. Đây là một kỹ thuật học máy không giám sát, giúp giảm kích thước của tập dữ liệu trong khi vẫn giữ lại nhiều thông tin nhất có thể. Đồng thời để mô hình có thể tận dụng được tất cả dữ liệu, chúng ta sử dụng kỹ thuật Cross-validation để huấn luyện mô hình.



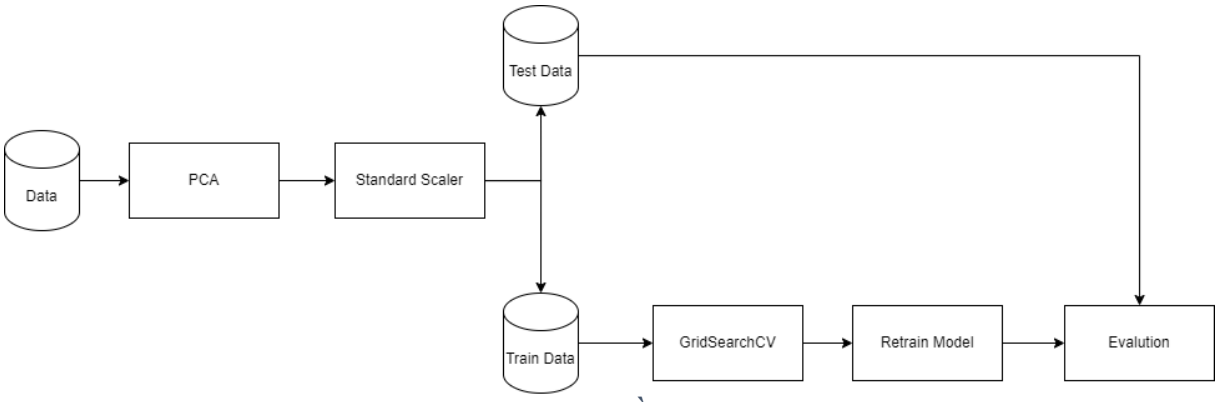
Hình 16 Mô tả Cross - Validation

Theo đó, tập Training data sẽ được chia thành những phần bằng nhau (như trong hình) và lấy một phần trong đó để đánh giá, các phần còn lại sẽ được dùng để huấn luyện mô hình. Như vậy, mô hình sẽ được học tối đa các trường hợp có trong dữ liệu, tận dụng được hết khả năng của tập dữ liệu. Ngoài ra, để tìm tham số cho mô hình, phương pháp GridSearchCV sẽ được sử dụng.



Hình 17 Mô tả GridSearchCV

Đối với mỗi tham số cần thiết để xây dựng mô hình, chúng ta sẽ cho đưa ra một vài giá trị khác nhau. GridSearchCV sẽ duyệt qua tất cả các mô hình có thể, đồng thời huấn luyện theo Cross-Validation để chọn ra những tham số cho kết quả cao nhất. Trong bài này, chúng ta sẽ sử dụng thuật toán Random Forest để xây dựng model, nhưng trước khi xây dựng model, chúng ta sẽ phải đưa tất cả các giá trị trong tập dữ liệu về trong khoảng từ -1 đến 1. Thao tác này gọi là chuẩn hoá dữ liệu, bởi các mô hình Học máy hoạt động tốt hơn với dữ liệu trong khoảng này. Sau đó, dữ liệu cần được chia thành tập Train Data và Test Data. Do đó, sơ đồ của quá trình xây dựng mô hình sẽ như sau:



Hình 18 Luồng xử lý

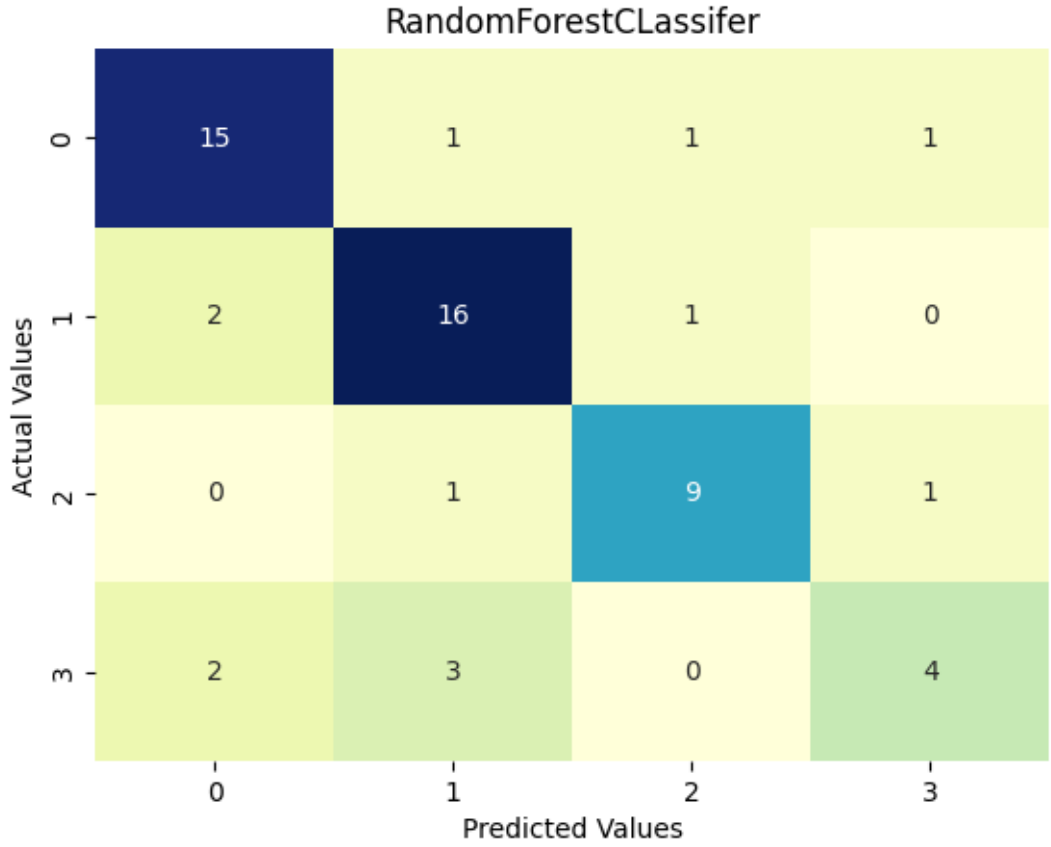
3.3 Đánh giá kết quả

Mô hình sau khi được huấn luyện bằng tập Train Data sẽ được đánh giá bằng cách dự đoán và so sánh kết quả trên tập Test Data. Kết quả của mô hình Random Forest trong bài này như sau:

Random Forest Accuracy Score: 77.19%

	precision	recall	f1-score	support
0	0.79	0.83	0.81	18
1	0.76	0.84	0.80	19
2	0.82	0.82	0.82	11
3	0.67	0.44	0.53	9
accuracy			0.77	57
macro avg	0.76	0.73	0.74	57
weighted avg	0.77	0.77	0.76	57

Hình 19 Kết quả tham số đánh giá



Hình 20 Ma trận nhầm lẫn (confusion matrix)

Từ hình trên ta thấy, chỉ số accuracy (độ chính xác) rơi khoảng 77 %, tương đối ổn. Tuy nhiên, khi nhìn vào các nhãn, ta có thể thấy các chỉ số của nhãn 3 (over_4_hours) tương đối thấp. Trong confusion matrix, việc dự đoán các nhãn có vẻ vẫn mang tính ngẫu nhiên tương đối cao. Tuy nhiên, với kết quả như thế này thì chưa đủ khả năng để có thể áp dụng vào thực tế.

Giải pháp hiện tại của bài toán này theo em đó là cần phải thu thập thêm một lượng dữ liệu thực tế nữa. Để mô hình có thể hiệu quả hơn, điều quan trọng nhất vẫn là phải có đầy đủ dữ liệu. Lượng dữ liệu của chúng ta hiện nay vẫn còn quá khiêm tốn. Việc sử dụng các phương pháp như nội suy hay giảm chiều dữ liệu

mặc dù có hiệu quả nhưng không thể hiệu quả bằng so với việc có thêm nhiều dữ liệu thực tế.

KẾT LUẬN CHUNG

Sau một thời gian ngắn được thực tập và tham gia các dự án tại phòng Kỹ thuật hệ thống của Công ty FPT Telecom, em đã rút ra được những kết quả sau đây.
Về những nội dung thực tập tại đơn vị:

- Tìm hiểu về cách vận hành và thiết kế của hệ thống FTTH
- Trải nghiệm những khó khăn trong xử lý những yêu cầu mang tính thực tế.
- Trải nghiệm tự mình xử lý một tập dữ liệu thô thực tế.

Do thời gian nghiên cứu có hạn nên các kết quả trong bài báo cáo này có kết quả không quá tốt, vì thế em mong nhận được ý kiến đóng góp từ thầy cô và các anh chị trong phòng Kỹ thuật hệ thống.
Một lần nữa em xin chân thành cảm ơn Cô Phạm Thị Thuý Hiền cũng như phía đơn vị Công ty Cổ phần Viễn thông FPT Telecom đã tạo điều kiện và giúp đỡ em rất nhiều trong khoảng thời gian thực tập này.

Em xin chân thành cảm ơn!