HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

VŨ VĂN ĐAM

DỰ BÁO KHÁCH HÀNG RỜI MẠNG DỊCH VỤ FIBERVNN TẠI VNPT NAM ĐỊNH

Chuyên ngành: Khoa học máy tính

Mã số: 8.48.01.01

TÓM TẮT ĐỀ ÁN TỐT NGHIỆP THẠC SĨ

HÀ NỘI – NĂM 2025

Đề án tốt nghiệp được hoàn thành tại:

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Người hướng dẫn khoa học: TS. Phan Thị Hà

Phản biện 1: ……..

Phản biện 2: ……..

Đề án tốt nghiệp sẽ được bảo vệ trước Hội đồng chấm đề án tốt nghiệp thạc sĩ tại Học viện Công nghệ Bưu chính Viễn thông

Vào lúc: ..... giờ ..... ngày ..... tháng ..... năm .....

Có thể tìm hiểu đề án tốt nghiệp tại:

- Thư viện của Học viện Công nghệ Bưu chính Viễn thông

MỞ ĐẦU

Như Trong bối cảnh thị trường viễn thông gần như đã bước vào giai đoạn bão hòa, việc phát triển thuê bao mới gặp nhiều khó khăn khi phần lớn khách hàng mới đến từ nhóm chuyển đổi từ các nhà mạng đối thủ. Do đó, chi phí để phát triển một khách hàng mới thường cao hơn đáng kể so với chi phí cần thiết để duy trì và giữ chân khách hàng hiện tại. Vì vậy, công tác chăm sóc và giữ chân khách hàng luôn được các nhà mạng, trong đó có VNPT Nam Định, xác định là nhiệm vụ trọng tâm.

Xuất phát từ thực tế đó, trong đề án này tôi lựa chọn áp dụng các phương pháp học máy để phân tích và dự báo khả năng rời mạng của khách hàng sử dụng dịch vụ FiberVNN tại VNPT Nam Định. Đây cũng chính là lý do tôi chọn đề tài nghiên cứu:

“ **Dự báo khách hàng rời mạng dịch vụ FiberVNN tại VNPT Nam Định”**

Nội dung của đề án được trình bày trong ba chương nội dung chính như sau:

Chương 1: Cơ sở lý luận

Trong chương này sẽ trình bày tổng quan về VNPT Nam Định, tổng quan về học máy và nghiên cứu một số thuật toán học máy áp dụng cho bài toán dự báo.

Chương 2: Phương pháp dự báo khách hàng rời mạng

Chương này sẽ giới thiệu tổng quan về phương pháp dự báo và xây dựng mô hình dự báo khách hàng rời mạng.

Chương 3: Cài đặt và thử nghiệm

Chương này sẽ trình bày về cài đặt môi trường thử nghiệm, cài đặt mô hình thử nghiệm và đánh giá kết quả dự báo.

Chương 1. Cơ sở lý luận

1.1 Tổng quan về VNPT Nam Định

1.1.1 Giới thiệu.

VNPT Nam Định là đơn vị hạch toán phụ thuộc Tập đoàn Bưu chính Viễn thông Việt Nam (VNPT), tiền thân là Bưu điện tỉnh Nam Định được thành lập từ năm 1945. Đến tháng 01 năm 2008, Bưu điện tỉnh Nam Định được chia tách thành hai đơn vị: Viễn thông Nam Định và Bưu điện Nam Định.

VNPT Nam Định có chức năng nhiệm vụ kinh doanh và cung cấp các dịch vụ viễn thông - công nghệ thông tin (VT - CNTT) trên địa bàn tỉnh, phục vụ nhu cầu thông tin liên lạc của các cấp ủy Đảng, chính quyền, cơ quan, đoàn thể, doanh nghiệp và nhân dân tỉnh Nam Định.

1.1.2 Tổng quan về dịch vụ Internet tại Nam Định.

Thị phần cụ thể của từng nhà cung cấp tính đến cuối năm 2023 như sau:

- VNPT chiếm khoảng 38% thị phần
- Viettel chiếm khoảng 41% thị phần
- FPT chiếm khoảng 18% thị phần
- Các nhà cung cấp khác chiếm khoảng 3% thị phần trên toàn tỉnh.

Bảng 1.1: Chi phí phát triển một khách hàng mới tại VNPT Nam Định

STT	Loại chi phí	Chi phí
1	Chi phí lắp đặt + hoa hồng	- Lắp đặt: 100.000đ /1 Khách hàng - Hoa hồng: 100.000đ/1 Khách hàng
2	Dây cáp quang (50 -100 mét)	400.000đ / 1 cuộn (1000 mét)
3	Dây mạng LAN(10 mét)	5.000đ/ 1 mét
4	Modem	800.000đ - 1.200.000đ
5	Chi phí hòa mạng	300.000đ
Tổng chi phí		**1.370.000đ**

1.2 Tổng quan về học máy

Các phương pháp học máy:

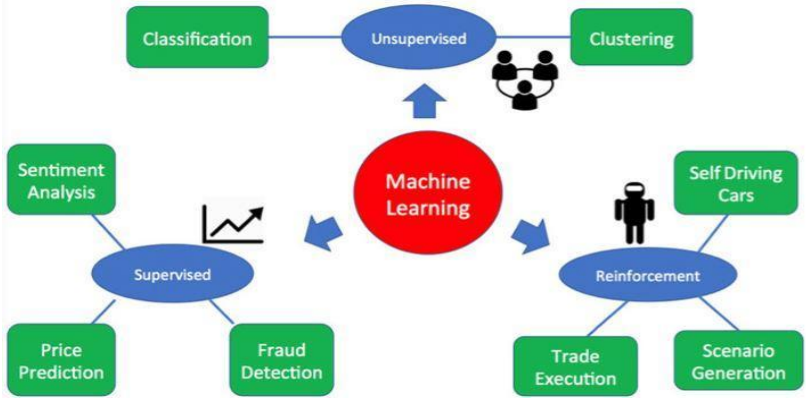
Học máy được chia thành ba nhóm phương pháp chính:

- Học có giám sát (supervised learning): Đây là phương pháp trong đó tập dữ liệu huấn luyện gồm các ví dụ kèm giá trị đầu ra. Mục tiêu là xây dựng mô hình từ dữ liệu huấn luyện để dự đoán chính xác giá trị đầu ra cho dữ liệu mới.

- Học không giám sát (un-supervised learning): Ở phương pháp này, tập dữ liệu chỉ gồm các ví dụ không kèm giá trị đầu ra. Thuật

toán học máy dựa trên độ tương tự giữa các ví dụ để phân nhóm thành các cụm, sao cho các ví dụ trong cùng cụm có tính chất tương đồng.

- Học tăng cường (reinforcement learning): Trong phương pháp này, hệ thống học qua tương tác với môi trường mà không được cung cấp trực tiếp đầu ra đúng. Hệ thống nhận phần thưởng (reward) sau mỗi chuỗi hành động và học cách chọn hành động để tối đa hóa tổng phần thưởng. Ví dụ: trong cờ vua, hệ thống chỉ biết kết quả cuối cùng (thắng/thua) để tự học chiến lược chơi hiệu quả.



Hình 1: Các phương pháp học máy

1.3 Nghiên cứu một số thuật toán học máy áp dụng cho bài toàn dự báo.

1.3.1. Học cây quyết định[3]

Cây quyết định (*decision tree*) là cấu trúc dạng cây dùng trong phân loại và hồi quy, nhận đầu vào là bộ giá trị thuộc tính mô tả đối tượng và trả về nhãn phân loại rời rạc. Mỗi bộ thuộc tính gọi là mẫu (sample), đầu ra là nhãn phân loại (label).

Thuật toán học cây quyết định

Trong đề tài này, ta tập trung vào thuật toán ID3 một trong những thuật toán học cây cơ bản, đại diện cho cách tiếp cận xây dựng cây theo hướng phân chia tuần tự. ID3 do Ross Quinlan phát triển và sau này được cải tiến thành thuật toán C4.5, phổ biến hơn trong thực tế.

1.3.1.1. Thuật toán ID3.

Thuật toán ID3 xây dựng cây quyết định từ dữ liệu huấn luyện nhằm phân loại đầu ra phù hợp với nhãn có sẵn. Do số lượng cây quá lớn, ID3 dùng chiến lược tham lam, xây dựng cây từ trên xuống và chọn thuộc tính tốt nhất tại mỗi bước.

Quá trình xây dựng cây diễn ra như sau:

Chọn thuộc tính cho nút gốc: Lựa chọn thuộc tính giúp phân chia dữ liệu thành các tập con đồng nhất (các mẫu cùng nhãn).

Phân chia dữ liệu: Tập dữ liệu được chia thành các nhánh con dựa trên giá trị của thuộc tính vừa chọn.

Đệ quy: Với mỗi tập con, thuật toán lặp lại quá trình chọn thuộc tính, loại bỏ các thuộc tính đã sử dụng, và tiếp tục xây cây.

Quá trình dừng khi:

Tập dữ liệu con tại một nút có cùng nhãn → gán nhãn đó cho nút lá.

Không còn thuộc tính để phân chia, nhưng dữ liệu vẫn còn nhiều nhãn → gán nhãn chiếm đa số cho nút.

Lựa chọn thuộc tính tốt nhất.

ID3 sử dụng entropy để đo mức độ không đồng nhất của dữ liệu và tính information gain nhằm chọn thuộc tính phân chia tốt nhất.

Với bài toán phân loại nhị phân (+ và -), entropy của tập dữ liệu S được tính để xác định thuộc tính giúp tăng độ đồng nhất nhiều nhất tại mỗi bước phân chia.

Entropy của tập dữ liệu S được tính theo công thức:

$$H(S) = -p^{+}\backslash log_2 p^{+} - p^{-}\backslash log_2 p^{-}$$

trong đó p+ và p- là xác suất quan sát thấy nhãn phân loại + và -, được tính bằng tần suất quan sát thấy + và – trong tập dữ liệu.

Trong trường hợp tổng quát với C nhãn phân loại có xác suất lần lượt là $p_1, p_2, \ldots, p_C$ entropy được tính như sau:

$$H(S) = -\sum_{i=1}^{C} p_i \log_2 p_i$$

Để lựa chọn thuộc tính phân tách tại mỗi nút của cây, ta xét độ tăng thông tin (Information Gain), ký hiệu là IG.

$$IG(S,A) = H(S) - \sum_{v \in values(A)} \frac{|S_V|}{|S|} H(S_V)$$

Trong đó:

S là tập dữ liệu ở nút hiện tại

A là thuộc tính

values(A) là tập các giá trị của thuộc tính A.

S_V là tập các mẫu có giá trị thuộc tính A bằng v.

|S| và $|S_V|$ là lực lượng của các tập hợp tương ứng.

1.3.1.2. Thuật toán C4.5

C4.5 là thuật toán xây dựng cây quyết định do J. R. Quinlan phát triển từ ID3 (1993), nổi bật nhờ khả năng xử lý linh hoạt, hiệu

quả cao

Đặc điểm chính của thuật toán C4.5:

- Sử dụng **Gain Ratio** thay Information Gain để chọn thuộc tính phân chia.
- Xử lý tốt thuộc tính rời rạc, liên tục và dữ liệu thiếu giá trị.
- Có cơ chế cắt tỉa cây để giảm overfitting.

Ý nghĩa của Gain Ratio (GR):

$$H_D(C) = -\sum_{k=1}^{k} p_k log_2(p_k) = -\sum_{k=1}^{k} \frac{|D_k|}{|D|} log_2\left(\frac{|D_k|}{|D|}\right)$$

$$H_D(C|F_i) = \sum_{j=1}^{p_i} \frac{|D_k|}{|D|} H_{D_J}\left(C|F_i = V_j^i\right)$$

$$IG_D(C|F_i) = H_D(C) - H_D(C|F_i)$$

Splitting entropy của thuộc tính F_i, ký hiệu $SE_D(F_i)$:

$$SE_D(F_i) = -\sum_{j=1}^{P_i} \frac{|D_j|}{|D|} \log_2\left(\frac{|D_j|}{|D|}\right)$$

Khi đó, Gain Ratio ký hiệu $GR_D(C|F_i)$

$$GR_D(C|F_i) = \frac{IG_D(C|F_i)}{SE_D(F_i)}$$

Gain Ratio giúp khắc phục nhược điểm của Information Gain khi ưu tiên thuộc tính có nhiều giá trị bằng cách chuẩn hóa theo Splitting Entropy, phản ánh mức phân tán dữ liệu khi phân chia.

1.3.2. Thuật toán SVM (Support Vector Machine)

Support Vector Machine (SVM) [8] là thuật toán học máy có giám sát mạnh mẽ, dùng cho phân loại và hồi quy, do Vapnik và cộng sự đề xuất. SVM tìm siêu phẳng tối ưu để phân tách dữ liệu với biên (margin) lớn nhất giữa hai lớp.

Mục tiêu của SVM là tìm siêu phẳng tối ưu để phân tách dữ liệu thành các lớp khác nhau sao cho khoảng cách biên (margin) giữa hai lớp là lớn nhất.

- Siêu phẳng và phân tách tuyến tính

Giả sử bài toán phân loại nhị phân với tập huấn luyện:

$$\left\{\left(x_i, y_i\right) \mid x_i \in R^n,\ y_i \in \{-1,1\}\right\}_{i=1}^{m}$$

Một siêu phẳng trong không gian R^n được biểu diễn bởi:

$w \cdot x + b = 0$

Trong đó:

w là vector trọng số (vector pháp tuyến).

b là hằng số điều chỉnh độ lệch (bias).

x là vector đặc trưng.

Dữ liệu được phân tách tuyến tính nếu tồn tại một siêu phẳng sao cho :

$$y_i\left(w \cdot x_i + b\right) \geq 1, \forall_i$$

- Hàm mục tiêu: Tối đa hóa biên (Margin)

Khoảng cách từ một điểm x đến siêu phẳng là:

$$\frac{\left| w \cdot x + b \right|}{||w||}$$

SVM tìm siêu phẳng sao cho khoảng cách này được tối đa hóa, tương đương với bài toán tối ưu:

$$\min_{w,b} \frac{1}{2}||w||^2 \text{ với điều kiện: } y_i\left(w \cdot x_i + b\right) \geq 1$$

- Trường hợp không phân tách tuyến tính.

Trong thực tế, dữ liệu thường không hoàn toàn phân tách tuyến tính. SVM sử dụng soft margin để cho phép một số điểm vi phạm điều kiện phân lớp:

$$\min_{w,b,\xi} \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\xi_i \text{ với } y_i\left(w \cdot x_i + b\right) \geq 1 - \xi_i,\ \xi_i \geq 0$$

Tham số C>0 điều khiển mức độ chấp nhận sai số:

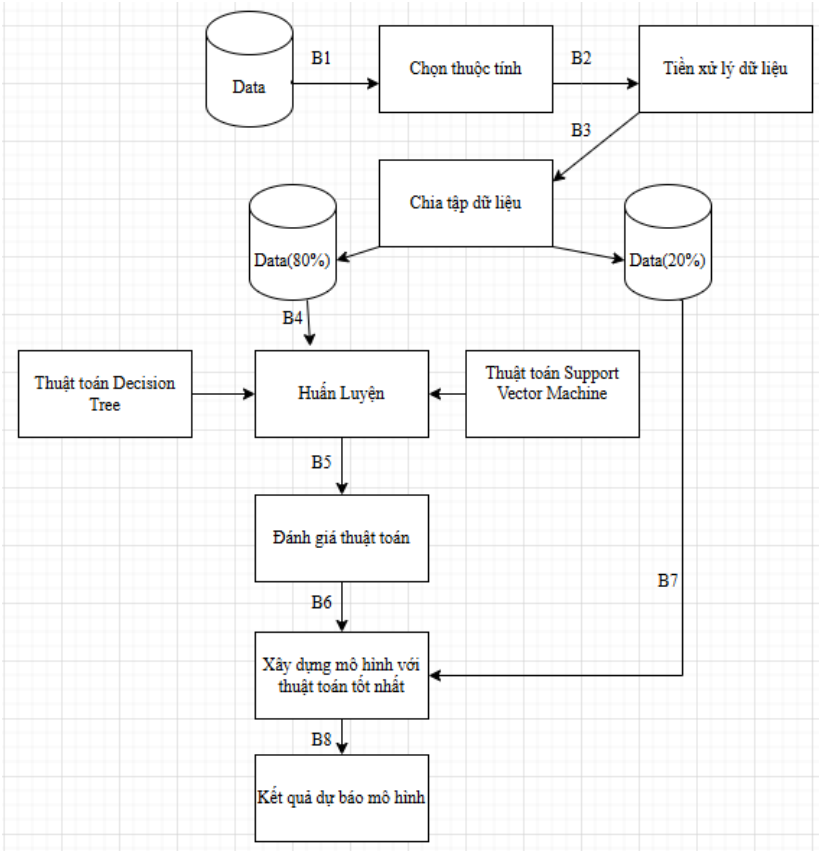
C lớn: ưu tiên phân loại chính xác (dễ overfit).

C nhỏ: chấp nhận một số lỗi để margin rộng hơn (regularization tốt hơn).

Chương 2. Phương pháp dự báo khách hàng rời mạng

2.1. Giới thiệu

Sơ đồ tổng quát hệ thống.



Hình : Sơ đồ tổng quát hệ thống

Luồng xử lý chính gồm 8 bước:

- B1: Lựa chọn thuộc tính
- B2: Tiền xử lý dữ liệu
- B3: Chia tập dữ liệu
- B4: Huấn luyện mô hình
- B5: Đánh giá thuật toán
- B6: Xây dựng mô hình với thuật toán tốt nhất

- B7: Kiểm thử
- B8: Kết quả

2.2. Xây dựng mô hình dự báo khách hàng rời mạng.

2.2.1. Thu thập và tiền xử lý dữ liệu.

Dữ liệu được truy xuất từ hệ thống nội bộ của VNPT Nam Định tới ngày 31/03/2025 bao gồm 140320 khách hàng đang và đã sử dụng dịch vụ Fiber.

- **Lựa chọn thuộc tính.**

Bằng kinh nghiệm, nghiệp vụ khách hàng cũng như kinh nghiệm xử lý dữ liệu tiến hành chọn lựa các thuộc tính và làm sạch dữ liệu. Các thuộc tính được lựa chọn để huấn luyện bao gồm 22 cột, cột 23 là cột Churn(thanhly), cột này là cột gắn nhãn của tập dữ liệu, cột để nhận biết là thuê bao có rời mạng hay không?

- **Xử lý dữ liệu:**

o Lọc dòng dữ liệu không hợp lệ: Đối với những dòng mà cột KHUVUC_ID trống hoặc LOAIKH_ID bằng 0 sẽ được loại bỏ để tránh ảnh hưởng tới kết quả dự báo.

o Điền các giá thị thiếu trong file Data, NGANHNGHE_ID=999 (khác).

LOAI_ID=1(Khác).

o Làm sạch trường tuổi khách hàng:

Với khách hàng có độ tuổi <= 10 -> giá trị được gán NaN, giá trị không hợp lệ.sử dụng thư viện numpy trong python để làm sạch trường tuổi không hợp lệ.

- Chia Data thành 3 nhóm chính:

selected_features: list[str] = numeric_features + categorical_features + [target]

Trong đó:

numeric_features: Là biến số

categorical_features: Là biến phân loại

target="THANHLY"

2.2.2. Lựa chọn thuật toán, Huấn luyện mô hình:

Đề án tập trung vào 2 thuật toán: Decision Tree và Support Vector Machine (SVM). Sau khi Training lựa chọn thuật toán tốt nhất để xây dựng mô hình dự báo. Một trong những chỉ số cơ bản và quan trọng nhất đó là:

- **Accuracy** (Độ chính xác):

$$Accuracy = \frac{\text{Số lượng mẫu dự đoán đúng}}{\text{Tổng số mẫu}} = \frac{TP + TN}{TP + TN + FP + FN}$$

- TP (True Positive): Số khách hàng rời mạng mô hình dự đoán đúng.
- FP (False Positive): Số khách hàng rời mạng mô hình dự đoán sai.
- TN (True Negative): Số khách hàng sử dụng mô hình dự đoán đúng.
- FN (False Negative): Số khách hàng sử dụng mô hình dự đoán sai.

- **Kappy** (Cohen's Kappa): Là một chỉ số thống kê dùng để đo

lường mức độ đồng thuận giữa hai bộ phân loại (hoặc giữa mô hình dự đoán và thực tế), có tính đến sự đồng thuận xảy ra ngẫu nhiên:

$$\kappa = \frac{P_o - P_e}{1 - P_e}$$

Trong đó:

- Po: Tỉ lệ quan sát đồng thuận thực tế (accuracy).
- Pe: Tỉ lệ đồng thuận ngẫu nhiên kỳ vọng.

- **MCC** (Matthews Correlation Coefficient): là hệ số tương quan giữa giá trị thực và giá trị dự đoán của mô hình

$$\text{MCC} = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Trong đó:

TP: Số khách hàng rời mạng mô hình dự đoán đúng

TN: Số khách hàng sử dụng mô hình dự đoán đúng.

FP: Số khách hàng rời mạng mô hình dự đoán sai.

FN: Số khách hàng sử dụng mô hình dự đoán sai.

Sử dụng PyCaret và Pandas giúp đơn giản hóa quy trình xử lý dữ liệu, xây dựng, huấn luyện và triển khai mô hình. Pandas xử lý, định nghĩa đặc trưng; PyCaret hỗ trợ thiết lập môi trường huấn luyện, tự động tiền xử lý, so sánh, chọn và lưu mô hình tốt nhất, đồng thời chuẩn bị pipeline hoàn chỉnh cho bài toán phân loại.

Tạo một mô hình Decision Tree và Support Vector Machine bằng PyCaret, huấn luyện nó trên dữ liệu đã qua setup()

```
print('********************Decisiontree**********************')
dt = create_model(estimator='dt', fold=5, return_train_score=True)
save_model(dt, 'churn-rate-decisiontree')
```

Hình 5: Lưu mô hình huấn luyện Decision Tree

```
print('********************Supportvectormachine*******************************')
svm = create_model(estimator='svm', fold=5, return_train_score=True)
save_model(svm, 'churn-rate-supportvectormachine')
```

Hình **7:** Lưu mô hình huấn luyện SVM

So sánh giữa hai mô hình sau khi huấn luyện.

```
********************So sanh Decisiontre
                          Model  Accuracy
dt    Decision Tree Classifier    0.9648
svm       SVM - Linear Kernel    0.5902

      Kappa     MCC  TT (Sec)
dt   0.0000  0.0000     2.864
svm  0.0014  0.0013     4.651
```


Hình 8: Kết quả so sánh giữa hai mô hình

⇨ Ta thấy thuật toán Decision Tree có hiệu suất đạt 96,48% và thời gian thực hiện huấn luyện là 2.864s. Thuật toán được lựa chọn để xây dựng là thuật toán Decision Tree.

- Xây dựng mô hình dự báo khách hàng rời mạng:

o Tải file CSV

```
# Upload file
uploaded_file = st.file_uploader("📂 Tải file CSV dữ liệu khách hàng:", type=["csv"])
```

Hình 9: Upload file CSV

o Đọc file CSV thành DataFrame + hiển thị bảng dữ liệu đầu vào.

```
    input_df = pd.read_csv(uploaded_file)
    st.subheader("📥 Dữ liệu đầu vào")
    st.dataframe(input_df)
```

Hình 10: Đọc file CSV

o Làm sạch và chuẩn bị dữ liệu: Xử lý dữ liệu (ví dụ: loại bỏ giá trị thiếu, chuẩn hóa), Nếu cột THANHLY tồn tại (label thực tế),

loại bỏ trước khi dự đoán.

```
cleaned_df = clean_data(input_df)
if 'THANHLY' in cleaned_df.columns:
    cleaned_df = cleaned_df.drop('THANHLY', axis=1)
```

Hình 11: Xử lý dữ liệu.

- Dự đoán bằng mô hình: Dùng mô hình đã load (hoặc train trước đó) để dự đoán.

```
predictions = predict_model(model, data=cleaned_df)
```

Hình 12: Dự đoán bằng mô hình.

- Ghép kết quả dự đoán vào dữ liệu gốc: Thêm cột dự đoán thanh lý và xác suất dự đoán vào bảng kết quả

```
# Ghép kết quả
full_result = input_df.copy()
label_col = next((col for col in predictions.columns if 'label' in col.lower()), None)
score_col = next((col for col in predictions.columns if 'score' in col.lower()), None)

if label_col:
    full_result["DU_DOAN_THANHLY"] = predictions[label_col]
if score_col:
    full_result["XAC_SUAT"] = predictions[score_col]
```

Hình 13: Ghép kết quả dự đoán vào dữ liệu gốc.

- So sánh với thực tế (nếu có cột THANHLY). Tạo cột SO_SANH để biết dự đoán đúng hay sai.

```
if 'THANHLY' in full_result.columns and label_col:
    full_result["SO_SANH"] = full_result.apply(
        lambda row: "✅ Đúng" if row["DU_DOAN_THANHLY"] == row["THANHLY"] else "❌ Sai", axis=1
    )
```

Hình 14: So sánh với thực tế.

- Hiển thị kết quả: Hiển thị bảng kết quả với dự đoán.

```
# ✅ Kết quả đầy đủ
st.subheader("📈 Kết quả dự đoán")
st.dataframe(full_result)
```

Hình 15: Hiển thị kết quả.

- Hiển thị và tô màu dòng sai nếu có: Tùy kích thước dữ

liệu, chỉ hiện dòng sai hoặc hiện toàn bộ và highlight dòng sai

```
# 🔎 So sánh nếu có thực tế
if "SO_SANH" in full_result.columns:
    st.subheader("🧪 So sánh dự đoán với thực tế")
```

Hình 16: Hiển thị tô màu dòng sai nếu có.

- Vẽ biểu đồ hình tròn: Vẽ biểu đồ hình tròn so sánh tỷ lệ đúng/sai.

```
# Tạo biểu đồ hình tròn
fig = px.pie(
    comparison_counts,
    names="Kết quả",
    values="Số lượng",
    color="Kết quả",
    color_discrete_map={"✅ Đúng": "green", "❌ Sai": "red"},
    title="Tỷ lệ dự đoán đúng và sai",
    hole=0.4  # Nếu muốn biểu đồ hình donut
)

st.plotly_chart(fig, use_container_width=True)
```

Hình 17: Biểu đồ so sánh tỷ lệ đúng/sai.

- Cho phép tải kết quả: Tải kết quả về dưới dạng CSV hoặc Excel.

```
# Tải kết quả CSV
csv_result = full_result.to_csv(index=False).encode('utf-8')
st.download_button("📥 Tải kết quả CSV", data=csv_result, file_name='du_doan_churn_rate.csv', mime='text/csv')

# Tải kết quả Excel
excel_buffer = BytesIO()
with pd.ExcelWriter(excel_buffer, engine='xlsxwriter') as writer:
    full_result.to_excel(writer, sheet_name="DuDoan", index=False)
st.download_button("📊 Tải kết quả Excel (.xlsx)", data=excel_buffer.getvalue(),
                   file_name='du_doan_churn_rate.xlsx',
                   mime='application/vnd.openxmlformats-officedocument.spreadsheetml.sheet')
```

Hình 18 : Trả kết quả dưới dạng CSV hoặc Excel

Chương 3. CÀI ĐẶT VÀ THỬ NGHIỆM

3.1. Cài đặt môi trường

Hệ thống được triển khai trên nền tảng máy chủ với cấu hình như sau:

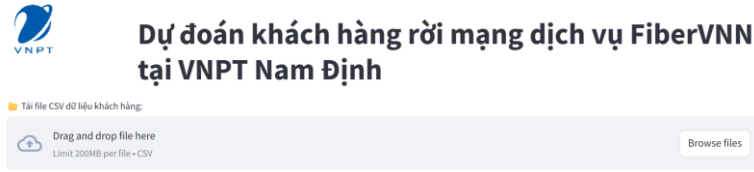
- Hệ điều hành: Windows Server 2019
- Bộ xử lý (Processor): Intel® Xeon® Gold 5318Y CPU @ 2.10 GHz (02 processors).
- Bộ nhớ RAM: 32 GB
- Hệ thống: 64-bit Operating System, x64-based processor

Môi trường lập trình sử dụng:

- Ngôn ngữ: Python 3.11
- Các thư viện chính được cài đặt và sử dụng: pandas, numpy, streamlit.

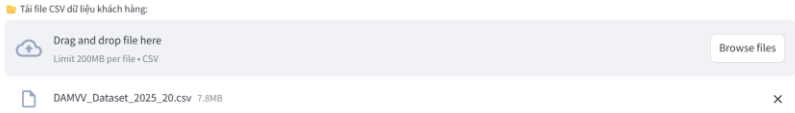
3.2. Thử nghiệm và đánh giá

Link truy cập: *https:// 7abc-113-175-171-43.ngrok-free.app*



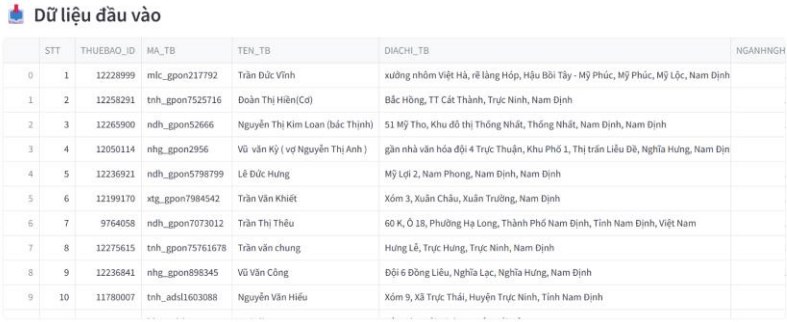
Hình 20: Giao diện ứng dụng dự đoán khách hàng rời mạng

-Chọn Browse files để chọn file khách hàng dự đoán:



Hình 21: Chọn file huấn luyện.

-Dữ liệu đầu vào để dự báo:



	STT	THUEBAO_ID	MA_TB	TEN_TB	DIACHI_TB	NGANHNGH
0	1	12228999	mlc_gpon217792	Trần Đức Vĩnh	xưởng nhôm Việt Hà, rẽ làng Hộp, Hậu Bồi Tây - Mỹ Phúc, Mỹ Phúc, Mỹ Lộc, Nam Định	
1	2	12258291	tnh_gpon7525716	Đoàn Thị Hiền(Cơ)	Bắc Hồng, TT Cát Thành, Trực Ninh, Nam Định	
2	3	12265900	ndh_gpon52666	Nguyễn Thị Kim Loan (bác Thịnh)	51 Mỹ Tho, Khu đô thị Thống Nhất, Thống Nhất, Nam Định, Nam Định	
3	4	12050114	nhg_gpon2956	Vũ văn Kỷ (vợ Nguyễn Thị Anh)	gần nhà văn hóa đội 4 Trực Thuận, Khu Phố 1, Thị trấn Liễu Đề, Nghĩa Hưng, Nam Định	
4	5	12236921	ndh_gpon5798799	Lê Đức Hưng	Mỹ Lợi 2, Nam Phong, Nam Định, Nam Định	
5	6	12199170	xtg_gpon7984542	Trần Văn Khiết	Xóm 3, Xuân Châu, Xuân Trường, Nam Định	
6	7	9764058	ndh_gpon7073012	Trần Thị Thêu	60 K, Ô 18, Phường Hạ Long, Thành Phố Nam Định, Tỉnh Nam Định, Việt Nam	
7	8	12275615	tnh_gpon75761678	Trần văn chung	Hưng Lễ, Trực Hưng, Trực Ninh, Nam Định	
8	9	12236841	nhg_gpon898345	Vũ Văn Công	Đội 6 Đồng Liêu, Nghĩa Lạc, Nghĩa Hưng, Nam Định	
9	10	11780007	tnh_adsl1603088	Nguyễn Văn Hiếu	Xóm 9, Xã Trực Thái, Huyện Trực Ninh, Tỉnh Nam Định	

Hình 22: Dữ liệu đầu vào.

-Kết quả dự báo:

	_KHL	SOLAN_TAMNGUNG	THANG_SD	KO_PSLL	SOLAN_GIAHAN	TRANGTHAITB_ID	TRANGTHAI_TB	THANHLY	DU_DOAN_THANHLY	XAC_SUAT	SO_SANH
0	0	0	11	0	2	1	Hoạt động bình thường	0	0	1	Đúng
1	0	0	6	0	0	1	Hoạt động bình thường	0	0	1	Đúng
2	0	0	4	0	0	1	Hoạt động bình thường	0	0	1	Đúng
3	0	0	41	0	3	1	Hoạt động bình thường	0	0	1	Đúng
4	0	0	10	0	2	1	Hoạt động bình thường	0	0	1	Đúng
5	0	0	17	0	2	1	Hoạt động bình thường	0	0	1	Đúng
6	0	0	85	0	6	1	Hoạt động bình thường	0	0	1	Đúng
7	0	0	1	0	2	1	Hoạt động bình thường	0	0	1	Đúng
8	0	0	10	0	2	1	Hoạt động bình thường	0	0	1	Đúng
9	0	0	85	0	0	1	Hoạt động bình thường	0	0	1	Đúng

Hình 23: Kết quả dự báo

-So sánh dự báo với thực tế: Chỉ hiển thị dòng báo sai

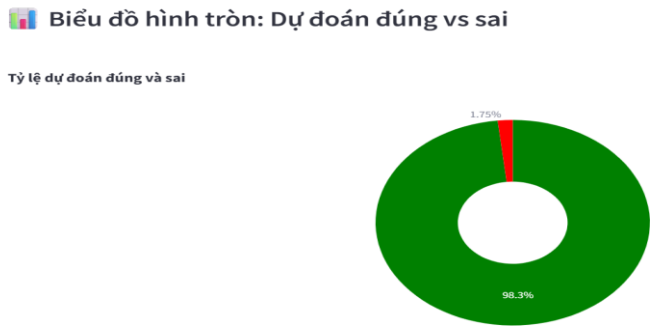
So sánh dự đoán với thực tế

⚠ Dữ liệu quá lớn. Chỉ hiển thị dòng dự đoán sai.

	KHL	SOLAN_TAMNGUNG	THANG_SD	KO_PSLL	SOLAN_GIAHAN	TRANGTHAITB_ID	TRANGTHAI_TB	THANHLY	DU_DOAN_THANHLY	XAC_SUAT	SO_SANH
56	0	0	15	0	0	1	Hoạt động bình thường	1	0	1	✗ Sai
200	0	1	11	0	0	6	Tạm dừng	1	0	1	✗ Sai
202	0	0	16	0	0	1	Hoạt động bình thường	1	0	1	✗ Sai
237	0	1	107	0	0	6	Tạm dừng	1	0	1	✗ Sai
253	0	0	22	0	0	1	Hoạt động bình thường	1	0	1	✗ Sai
263	0	1	16	0	0	6	Tạm dừng	1	0	1	✗ Sai
278	0	0	25	0	3	1	Hoạt động bình thường	1	0	1	✗ Sai
297	0	1	28	0	2	6	Tạm dừng	1	0	1	✗ Sai
347	0	0	11	0	0	1	Hoạt động bình thường	1	0	1	✗ Sai

Hình 24: So sánh dự báo rời mạng với thực tế

-Biểu đồ hình tròn, tỷ lệ dự báo đúng sai: Màu xanh là đúng, màu đỏ là sai



Hình 25: Biểu đồ dự báo đúng sai

Tỉ lệ dự báo đúng đạt 98,3% , tỉ lệ dự báo sai là 1,75%.

KẾT LUẬN

Kết quả đạt được

Đề án này đã đạt được một số kết quả sau:

- Thu thập và xây dựng cơ sở dữ liệu khách hàng, bao gồm các thông tin đặc tả như: đối tượng khách hàng, loại hình khách hàng, gói cước sử dụng, tình trạng nợ cước, số lượng dịch vụ đang sử dụng và các đặc điểm liên quan khác.
- Rút trích các thuộc tính đặc trưng, xác định các yếu tố có khả năng ảnh hưởng đến nguy cơ rời mạng của khách hàng, qua đó xây dựng tập dữ liệu huấn luyện phục vụ cho việc xây dựng và đánh giá mô hình dự báo.
- Tiền xử lý dữ liệu và xây dựng mô hình dự báo có độ chính xác cao, cụ thể xây dựng mô hình dự báo với thuật toán Decision Tree độ chính xác 96.48%.

Hướng phát triển

- Hoàn Tiếp tục nghiên cứu và thử nghiệm các mô hình bằng các thuật toán như Random Forest/ Gradient Boosting / XGBoost / LightGBM / CatBoost, Logistic Regression, K-Nearest Neighbors (KNN), Neural Network (Mạng nơ-ron nhân tạo) Để tìm ra thuật toán tối ưu nhất cho bài toán.
- Dự báo và thử nghiệm trên các dịch vụ khác của VNPT Nam Định như MyTV, Điện thoại di động…
- Gửi thông báo đến nhân viên quản lý địa bàn về thông tin khách hàng có nguy cơ rời mạng để tiến hành chăm sóc và thuyết phục khách hàng tiếp tục sử dụng dịch vụ của đơn vị.

DANH MỤC TÀI LIỆU THAM KHẢO

[1] Dương Minh Lý, (2021), Luận văn Thạc sĩ Dự báo Khách hàng sử dụng dịch vụ FiberVNN của VNPT Tây Ninh có nguy cơ rời mạng, Học viện Công Nghệ Bưu Chính Viễn Thông cơ sở TP.HCM

[2] Nguyễn Thị Như Ngọc, (2014), Luận văn Thạc sĩ Phân tích dữ liệu thuê bao di động hướng đến dự báo thuê bao rời mạng viễn thông, Trường Đại học Công nghệ – Đại học Quốc gia Hà Nội

[3] T.M.Phương, Giáo trình Nhập môn trí tuệ nhân tạo, Hà Nội: Học viện Công nghệ Bưu chính Viễn Thông, 2015

[4] Võ Đức, V., & Trần, V. L. (2022). Dự Báo Rời Mạng Dịch Vụ Fiber. Tạp Chí Khoa học HUFLIT, 7(1), 3.

[5] Số liệu kinh doanh của VNPT Nam Định được truy xuất vào 31/03/2025

[6] Ionut Cortes, C., & Vapnik, V. (1995). *Support-vector networks*. Machine learning, 20(3), 273-297.

[7] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

[8] J. Ross Quinlan. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, 1993

[9] Wu, Lin and Weng, "*Probability estimates for multi-class classification by pairwise coupling*", JMLR 5:975-1005, 2004.

[10] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[11] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 3rd ed. O'Reilly Media, 2022.

[12] Scikit-learn Developers, "Scikit-learn Documentation," 2024. [Online]. Available: https://scikit-learn.org/stable/

[13] Scikit-learn Developers, "Decision Trees — Scikit-learn Documentation," 2024. [Online]. Available: https://scikit-learn.org/stable/modules/tree.html

[14] Scikit-learn Developers, "Support Vector Machines — Scikit-learn Documentation," 2024.