**BÁO CÁO CHẤM ĐIỂM TÍN DỤNG CỦA KHÁCH HÀNG**

# GIỚI THIỆU

# NGHIÊN CỨU LIÊN QUAN

# PHƯƠNG PHÁP

## 3.1. Các kỹ thuật nền tảng

Trong phần này, báo cáo sẽ trình bày và giải thích các kỹ thuật cơ bản để xây dựng một hệ thống chấm điểm tín dụng thường sử dụng. Các kỹ thuật này là các công cụ và phương pháp chính để xây dựng mô hình dự đoán rủi ro tín dụng và đánh giá khả năng trả nợ của khách hàng. Dưới đây là một số kỹ thuật quan trọng:

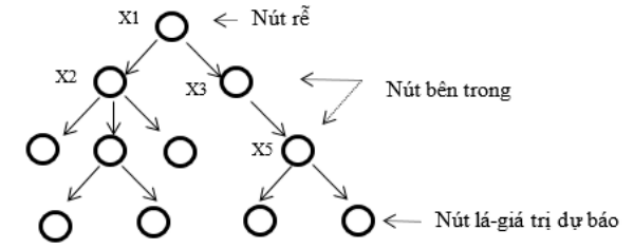
* *Phân tích và xử lý dữ liệu:* Đây là bước quan trọng nhất trong quá trình xây dựng mô hình chấm điểm tín dụng. Phân tích dữ liệu giúp hiểu rõ hơn về các đặc điểm của khách hàng và mối quan hệ giữa các yếu tố dữ liệu và khả năng trả nợ của họ. Xử lý dữ liệu bao gồm việc xử lý các giá trị bị thiếu, chọn lọc đặc trưng và chuẩn hóa dữ liệu bằng Min-Max Scaling (chuẩn hóa Min-Max) hay Standard Scaler.
* *Xử lý mất cân bằng dữ liệu:* Trong các vấn đề chấm điểm tín dụng, thường có sự mất cân bằng giữa các nhóm dữ liệu, ví dụ như số lượng khách hàng trả nợ và không trả nợ. Các kỹ thuật như Oversampling, Undersampling, và SMOTE được sử dụng để giải quyết vấn đề này.
* *Mô hình hóa dữ liệu:* Các kỹ thuật này bao gồm việc lựa chọn và xây dựng các mô hình phân loại và học máy để dự đoán rủi ro tín dụng. Các mô hình phổ biến bao gồm Logistic Regression, Decision Trees, Random Forest, Adaboost, và Neural Networks...
* *Đánh giá mô hình:* Kỹ thuật này liên quan đến việc đánh giá hiệu suất của các mô hình dự đoán. Các phương pháp đánh giá bao gồm Confusion Matrix, ROC Curve, và các độ đo như Accuracy, Precision, Recall và F1-score.
* *Tinh chỉnh mô hình:* Sau khi xây dựng mô hình, quá trình tinh chỉnh là cần thiết để điều chỉnh các tham số mô hình và tối ưu hóa hiệu suất của nó. Các kỹ thuật như Cross-Validation và Grid Search được sử dụng để tinh chỉnh mô hình.

Các kỹ thuật nền tảng này cùng với các công cụ và phương pháp khác tạo ra cơ sở cho việc phát triển các mô hình chấm điểm tín dụng hiệu quả và chính xác. Trong phần tiếp theo, chúng ta sẽ đề xuất phương pháp cụ thể để áp dụng các kỹ thuật này vào việc giải quyết vấn đề chấm điểm tín dụng khách hàng.

## 3.2. Đề xuất phương pháp

Vấn đề lớn đối với các tổ chức tài chính và ngân hàng là xác định tín dụng xấu, vì nó có thể gây ra các vấn đề nghiêm trọng trong tương lai như thất thoát vốn, giảm doanh thu và tăng tổn thất cho ngân hàng. Hiện nay, có rất nhiều kĩ thuật học máy để chấm điểm tín dụng của khách hàng. Báo cáo đã đề xuất sử dụng một tổ hợp các mô hình học máy truyền thống và mô hình học sâu để chấm điểm tín dụng của khách hàng. Cụ thể, báo cáo đề xuất sử dụng 7 mô hình sau đây:

* *Cây quyết định (Decision Tree - DT):* DT là một thuật toán học máy được ứng dụng rộng rãi trong phân loại và dự đoán, bao gồm cả lĩnh vực đánh giá rủi ro tín dụng. Mô hình này hoạt động dựa trên hệ thống các quy tắc logic được biểu diễn dưới dạng cây, giúp phân loại khách hàng và dự đoán khả năng thanh toán khoản vay của họ. Mô hình này bao gồm một nút gốc (Root node), các nút bên trong (Internal node) và nút lá (Leaf node). Cấu trúc của cây quyết định gồm nút gốc - đại diện cho điểm khởi đầu của quá trình phân loại, nút bên trong - thể hiện các thuộc tính (biến) được sử dụng để phân chia dữ liệu và nút lá - đại diện cho kết quả dự đoán (ví dụ: mức độ rủi ro tín dụng) (Hình 1).



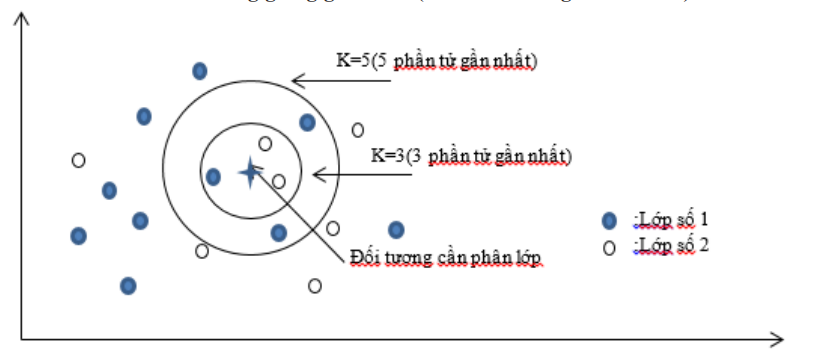
**Hình 1.** Cây quyết định (Decision tree)

*Nguồn: Abdou. (2011) [5]*

* *K-Láng giềng gần nhất (K-Nearest Neighbor - KNN):* KNN là một mô hình dựa trên sự gần gũi của điểm dữ liệu, dựa vào các điểm láng giềng gần nhất để dự đoán nhãn của một điểm dữ liệu mới.

Từ Hình 2 có thể thấy, nếu chọn số điểm gần nhất để phân loại là 3 thì xung quanh điểm cần phân lớp có 3 điểm trong đó 1 điểm thuộc lớp 1 và 2 điểm thuộc lớp 2. Với số điểm thuộc lớp 2 nhiều hơn, điểm cần phân lớp được đưa vào lớp số 1. Tương tự như vậy, nếu lựa chọn số điểm gần nhất k=5 thì có năm điểm xung quanh điểm cần phân lớp, trong đó có 3 điểm thuộc lớp số 1 và 2 điểm thuộc lớp 2. Nếu theo dữ liệu này thì điểm cần phân lớp sẽ thuộc lớp số 1. Trên thực tế dữ liệu cần phân loại có nhiều thuộc tính trong đó mỗi thuộc tính tương ứng với một chiều không gian, do vậy khi tính khoảng cách gần nhất cần tính khoảng cách vector trong không gian đa chiều với công thức khoảng cách Euclidean:

trong đó: x, y là 2 điểm cần xác định khoảng cách; n là số chiều không gian (số thuộc tính của biến mục tiêu).



**Hình 2.** K láng giềng gần nhất (K-Nearest Neighbors - KNN)

*Nguồn: Marinakis và cộng sự (2008) [6]*

* *Rừng ngẫu nhiên (Random Forest - RF)*: RF là một mô hình ensemble của nhiều cây quyết định, có khả năng tạo ra các dự đoán mạnh mẽ và ổn định bằng cách kết hợp kết quả từ nhiều cây quyết định. Random Forest có thể được sử dụng để dự đoán khả năng thanh toán của khách hàng dựa trên các đặc trưng tài chính và hành vi thanh toán. Điều này là do RF thích hợp với việc xử lý dữ liệu có cấu trúc và có khả năng làm việc với các đặc trưng có mức độ quan trọng khác nhau mà không cần phải tiền xử lý đặc biệt.
* *XGBoost (eXtreme Gradient Boosting - XGB):* XGB là một mô hình ensemble learning, cải tiến từ Gradient Boosting, giúp cải thiện hiệu suất dự đoán bằng cách tối ưu hóa hàm mất mát và xây dựng các cây quyết định tuần tự.
* *Naive Bayes (NB):* Naive Bayes là một mô hình đơn giản dựa trên xác suất, dựa trên giả định về sự độc lập giữa các biến, nhưng thường cho kết quả tốt trong nhiều bài toán phân loại.
* *Mạng nơ-ron tích chập sâu (Convolutional Neural Network - CNN):* CNN là một loại mạng nơ-ron nhân tạo đặc biệt phù hợp để phân tích dữ liệu dạng lưới, phổ biến nhất là hình ảnh. Nó có khả năng học một lượng lớn các dữ liệu trong khoảng thời gian ngắn hơn nhiều so với mạng nơ-ron thông thường. Lý do là nó sử dụng ít trọng số hơn trong khi độ chính xác chỉ kém hơn một phần nhỏ so với kiến trúc truyền thống.
* *Mạng nơ-ron hồi quy (Recurrent Neural Network - RNN):* RNN là một mô hình mạng neural thích hợp cho việc xử lý dữ liệu chuỗi như văn bản hoặc chuỗi thời gian, có khả năng nhớ thông tin từ quá khứ để đưa ra dự đoán tín dụng cho tương lai.

Sử dụng một tổ hợp của các mô hình này giúp tăng cường khả năng dự đoán và đảm bảo tính đa dạng trong quá trình đánh giá tín dụng của khách hàng. Các mô hình truyền thống cung cấp một cơ sở ổn định và dễ hiểu, trong khi các mô hình học sâu có khả năng học được các biểu diễn phức tạp từ dữ liệu, giúp cải thiện hiệu suất dự đoán trong các trường hợp phức tạp.

Phương pháp được thực hiện theo tiến trình:

**Bước 1:** Khai thác dữ liệu

* Quan sát tổng quan dữ liệu (Hệ tương quan, giá trị thống kê, vấn đề mất cân bằng dữ liệu ở biến mục tiêu).

**Bước 2:** Chuẩn bị dữ liệu

* Tiến hành tiền xử lý dữ liệu (xóa trùng lặp, kiểm tra dữ liệu trống), biến đổi các trường dữ liệu định danh với phương pháp mã hóa nhãn.
* Xóa các đặc trưng không quan trọng.
* Sử dụng phương pháp IQR để loại bỏ các giá trị ngoại lệ.
* Xử lý mất cân bằng dữ liệu.
* Sử dụng phương pháp Min-Max Scaling để chuẩn hóa.

**Bước 3:** Xây dựng tập dữ liệu huấn luyện và kiểm tra

Bộ dữ liệu được chia thành 2 tập gồm tập huấn luyện 80% và tập dữ liệu kiểm tra 20%.

**Bước 4:** Xây dựng và huấn luyện các mô hình

* Với tập dữ liệu huấn luyện tiến hành xây dựng và huấn luyện các mô hình.

**Bước 5:** Mô hình thử nghiệm với tập dữ liệu kiểm tra đưa ra đánh giá mô hình.

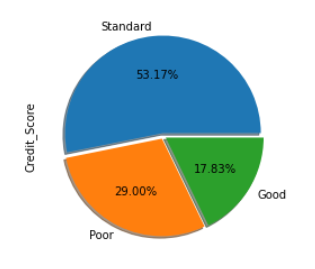
**Bước 6:** So sánh các mô hình dựa trên các chỉ số đánh giá hiệu suất và đưa ra mô hình tốt nhất.

**IV. THỰC NGHIỆM**

## 4.1. Miêu tả dữ liệu

Để tuân thủ các quy định pháp luật hiện hành về bảo mật thông tin cá nhân, các tổ chức tài chính phải tuân thủ nguyên tắc không tiết lộ thông tin chi tiết về tài khoản và các giao dịch của khách hàng. Trong phạm vi nghiên cứu này, chúng tôi tiến hành thực nghiệm trên một tập dữ liệu được công bố công khai có tên là “Credit score classification”, được truy cập và tải về từ trang Kaggle.com [3]. Bộ dữ liệu này được thiết kế để mô phỏng các loại dữ liệu giao dịch tài chính và không liên quan trực tiếp đến bất kỳ ngân hàng cụ thể nào. Tập dữ liệu bao gồm 28 đặc trưng với 100,000 quan sát và chi tiết được mô tả như trong Bảng 1.

Trường dữ liệu “Credit Score”: được sử dụng để đánh giá khả năng thanh toán của khách hàng và phản ánh lịch sử tín dụng của họ. Trường dữ liệu này có ba nhóm giá trị: “Good” (tốt), “Standard” (trung bình), và “Poor” (kém). Nhãn “Good” được gán cho những khách hàng có điểm tín dụng cao, thường là do họ có lịch sử thanh toán nợ đúng hạn và duy trì một hồ sơ tín dụng tích cực, nhãn này có 17828 quan sát chiếm 17.828%. Nhãn “Standard” thường ám chỉ đến những khách hàng có lịch sử tín dụng trung bình, có thể có một số vấn đề hoặc trễ hẹn trong quá khứ, nhãn này có 53174 quan sát chiếm 53.174%. Trong khi đó, nhãn “Poor” được gán cho những khách hàng có điểm tín dụng thấp, thường xuyên trễ hẹn hoặc không thực hiện thanh toán đúng hạn, nhãn này có 28998 quan sát chiếm 28.998%. Thông qua việc phân loại khách hàng thành các nhóm dựa trên biến "Credit\_Score", các tổ chức tài chính có thể chấm điểm tín dụng khách hàng dựa trên lịch sử tín dụng và các yếu tố liên quan.



**Hình 3.** Tỷ lệ các quan sát trong trường dữ liệu “Credit\_Score”

**Bảng 1:** Mô tả dữ liệu thực nghiệm

|  |  |  |
| --- | --- | --- |
| **Trường dữ liệu** | **Mô tả** | **Kiểu dữ liệu** |
| **ID** | Định danh duy nhất của mỗi quan sát | object |
| **Customer\_ID** | Định danh của khách hàng | object |
| **Month** | Tháng | object |
| **Name** | Tên khách hàng | object |
| **Age** | Tuổi khách hàng | object |
| **SSN** | Số An sinh Xã hội | object |
| **Occupation** | Nghề nghiệp của khách hàng | object |
| **Annual\_Income** | Thu nhập hàng năm của khách hàng | object |
| **Monthly\_Inhand\_Salary** | Lương thực nhận hàng tháng | float64 |
| **Num\_Bank\_Accounts** | Số tài khoản ngân hàng | int64 |
| **Num\_Credit\_Card** | Số thẻ tín dụng | int64 |
| **Interest\_Rate** | Lãi suất | int64 |
| **Num\_of\_Loan** | Số khoản vay | object |
| **Type\_of\_Loan** | Loại khoản vay | object |
| **Delay\_from\_due\_date** | Số ngày trễ so với ngày đáo hạn | int64 |
| **Num\_of\_Delayed\_Payment** | Số lượng thanh toán bị trễ | object |
| **Changed\_Credit\_Limit** | Số lượng thay đổi hạn mức tín dụng | object |
| **Num\_Credit\_Inquiries** | Số lượng điều tra tín dụng | float64 |
| **Credit\_Mix** | Tổ hợp tín dụng | object |
| **Outstanding\_Debt** | Nợ chưa thanh toán | object |
| **Credit\_Utilization\_Ratio** | Tỷ lệ sử dụng tín dụng | float64 |
| **Credit\_History\_Age** | Lịch sử tín dụng | object |
| **Payment\_of\_Min\_Amount** | Số tiền thanh toán tối thiểu | object |
| **Total\_EMI\_per\_month** | Tổng EMI hàng tháng | float64 |
| **Amount\_invested\_monthly** | Số tiền đầu tư hàng tháng | object |
| **Payment\_Behaviour** | Hành vi thanh toán | object |
| **Monthly\_Balance** | Số dư hàng tháng | object |
| **Credit\_Score** | Điểm tín dụng | object |

## 4.2. Tiền xử lý dữ liệu

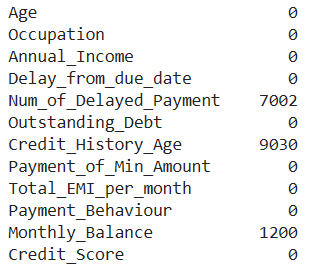
Việc tiền xử lý dữ liệu là một bước quan trọng trong quá trình chuẩn bị dữ liệu trước khi đưa vào quá trình huấn luyện mô hình học máy. Mục tiêu của tiền xử lý dữ liệu là tối ưu hóa dữ liệu để tăng cường hiệu suất của mô hình và cải thiện độ chính xác của mô hình theo mô hình học máy [1]. Chúng tôi đã thực hiện tiền xử lý dữ liệu với các bước như sau:

**Bước 1:** *Loại bỏ dữ liệu không cần thiết*

Loại bỏ các cột dữ liệu không sử dụng trong mô hình học máy, 16 trường dữ liệu này bị loại bỏ đi bao gồm là 'ID', 'Customer\_ID', 'Month', 'Name', 'SSN', 'Monthly\_Inhand\_Salary', 'Num\_Bank\_Accounts', 'Num\_Credit\_Card', 'Interest\_Rate', 'Num\_of\_Loan', 'Type\_of\_Loan', 'Changed\_Credit\_Limit', 'Num\_Credit\_Inquiries', 'Credit\_Mix', 'Credit\_Utilization\_Ratio', 'Amount\_invested\_monthly'

**Bước 2:** *Loại bỏ các giá trị NaN*

Xóa các hàng chứa giá trị NaN, cụ thể số lượng hàng bị xóa do các trường dữ liệu có chứa giá trị NaN được thể hiện trong hình 1.



**Hình 4.** Số lượng giá trị lỗi NaN trong các trường dữ liệu

**Bước 3:** *Loại bỏ các hàng chứa giá trị không chính xác và thay thế các dữ liệu sai*

Xóa các hàng có dữ liệu không chính xác trong cột "Occupation" và "Payment\_Behaviour", thay thế ký tự đặc biệt bằng chuỗi rỗng.

**Bước 4:** *Mã hóa và chuẩn hóa dữ liệu*

* Chuyển đổi dữ liệu cột "Credit\_History\_Age" từ chuỗi thành số, cụ thể thay thế chuỗi con ' Years and ' (bao gồm khoảng trắng) bằng dấu chấm ('.') và chuỗi con 'Months' (bao gồm khoảng trắng) bằng chuỗi rỗng (''). Ví dụ: '2 Years and 3 Months' thành '2.3'. Mục đích là chuyển đổi định dạng từ năm và tháng sang dạng thập phân để tính toán dễ dàng hơn.
* Chuyển đổi các nhãn trong cột “Payment\_Behaviour” thành các giá trị theo cấp độ. Sử dụng hàm ‘.str.replace(...)’ để thay thế gắn nhẵn “Low\_spent\_Small\_value\_payments” thành ‘1’,

“Low\_spent\_Medium\_value\_payments” thành ‘2’,

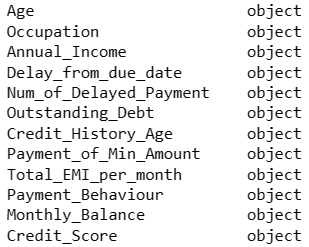
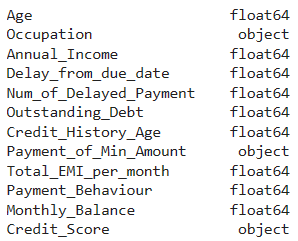
“Low\_spent\_Large\_value\_payments” thành ‘3’,

“High\_spent\_Small\_value\_payments” thành ‘4’,

“High\_spent\_Medium\_value\_payments” thành ‘5’,

“ High\_spent\_Large\_value\_payments” thành ‘6’.

* Chuyển đổi kiểu dữ liệu của các cột 'Age', 'Delay\_from\_due\_date', 'Num\_of\_Delayed\_Payment', 'Outstanding\_Debt', 'Total\_EMI\_per\_month', 'Monthly\_Balance', 'Payment\_Behaviour', 'Credit\_History\_Age', 'Annual\_Income' sang số thập phân

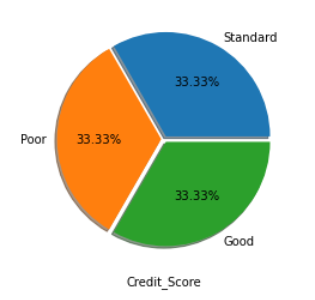
*Trước khi xử lý* *Sau khi xử lý*

**Hình 5.** Dữ liệu trước và sau khi thay đổi kiểu dữ liệu

* Tương tự đối với cột “Credit\_Score” và “Payment\_of\_Min\_Amount”, các nhãn của cột “Credit\_Score” lần lượt là ‘Good’, ‘Standard’, ‘Poor’ sẽ được thay thế thành ‘3’, ‘2’, ‘1’ và các nhãn cột “Payment\_of\_Min\_Amount” lần lượt là ‘NM’. ‘Yes’, ‘No’ được thay thế thành ‘0’, ‘1’, ‘2’, ‘3’. Cuối cùng chuyển kiểu dữ liệu 2 cột này thành số nguyên int64
* Sử dụng phương pháp IQR để loại bỏ các giá trị ngoại lệ. Đây là một phần quan trọng của tiền xử lý dữ liệu, vì ngoại lệ có thể làm méo mó các mô hình học máy và dẫn đến dự đoán không chính xác.

Phương pháp IQR (Interquartile Range) thường được sử dụng để xác định ngoại lệ. IQR là phạm vi giữa quartile thứ nhất (Q1) và quartile thứ ba (Q3) của tập dữ liệu. Các giá trị nằm ngoài khoảng từ Q1 - 1.5IQR đến Q3 + 1.5IQR thường được coi là ngoại lệ và có thể bị loại bỏ khỏi dữ liệu phân tích [8].

**Bước 5:** Sử dụng module Random Over Sampler xử lý sự mất cân bằng dữ liệu trong tập huấn luyện



**Hình 6.** Tỷ lệ các quan sát trong trường dữ liệu “Credit\_Score” sau khi cân bằng

**Bước 6:** Sử dụng phương pháp Min-Max Scaling để chuẩn hóa các biến số (numerical variables). Các trường dữ liệu thuộc biến độc lập lúc này sẽ được chuẩn hóa Min - Max đưa dữ liệu về một phân bố trong phạm vi [0, 1], chuẩn hóa được xác định theo Công thức (1).

(1)

trong đó: x là giá trị ban đầu

x\_scaled là giá trị sau khi áp dụng Min-Max Scaling (thường nằm trong khoảng 0 đến 1)

x\_min, x\_max lần lượt là giá trị nhỏ nhất, giá trị lớn nhất của trường dữ liệu trong tập dữ liệu.

Bảng 2 minh họa 3 quan sát đầu của dữ liệu sau khi được chuẩn hóa.

**Bảng 2:** Dữ liệu sau chuẩn hóa

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Age | Annual Income | Delay from due date | Num of Delayed Payment | Outstanding Debt | Credit History Age | Total EMI per month | Monthly Balance |
| 0.07031 | 0.083668 | 0.111111 | 0.002500 | 0.162020 | 0.67062 | 0.000602 | 1.0 |
| 0.10937 | 0.192389 | 0.166667 | 0.000909 | 0.121012 | 0.79228 | 0.000229 | 1.0 |
| 0.10937 | 0.192389 | 0.111111 | 0.000455 | 0.121012 | 0.79525 | 0.000229 | 1.0 |

**Bước 7:** Xử lý trường dữ liệu nghề nghiệp

* Sử dụng phương pháp One - Hot Encoding để biến đổi các giá trị của cột "Occupation" thành các biến giả định dạng số.

Kết quả là DataFrame chứa dữ liệu cần biến đổi sẽ bao gồm các biến giả định mới được tạo ra từ One-Hot Encoding, mỗi biến giả định tương ứng với một nhóm nghề nghiệp (Occupation) và chứa các giá trị 0 hoặc 1 để biểu thị sự hiện diện của từng nhóm nghề nghiệp [2].

**Bước 8:** *Xác định dữ liệu tập huấn và kiểm tra*

Bộ dữ liệu được chia theo tỷ lệ 80:20 với tập huấn luyện (80%) và tập kiểm tra (20%), việc huấn luyện mô hình trên toàn bộ tập dữ liệu có thể tốn nhiều thời gian và tài nguyên, đặc biệt với các tập dữ liệu lớn. Chia dữ liệu thành hai tập nhỏ hơn giúp giảm thời gian huấn luyện và cho phép bạn thử nghiệm nhiều mô hình khác nhau một cách hiệu quả hơn. Mục tiêu là để đảm bảo rằng dữ liệu được chia dẫn tới phương sai không quá cao [4].

* Tập huấn luyện (80%) cung cấp dữ liệu cho mô hình học và xây dựng mối quan hệ giữa các biến.
* Tập kiểm tra (20%) là tập dữ liệu độc lập, chưa được sử dụng trong quá trình huấn luyện. Việc đánh giá hiệu suất mô hình trên tập kiểm tra giúp đảm bảo tính khách quan, tránh tình trạng "quá khớp" (overfitting) - khi mô hình học thuộc lòng tập huấn luyện nhưng không thể khái quát hóa tốt cho dữ liệu mới.

## 4.3. Các độ đo đánh giá hiệu năng

Trong bài nghiên cứu này, các chỉ số đánh giá hiệu suất mà chúng tôi đã sử dụng là các chỉ số đánh giá Accuracy, Precision, Recall và F1 score [7], dựa trên ma trận nhầm lẫn có dạng như sau:

* Phương pháp Accuracy (độ chính xác) của mô hình là thước đo đơn giản nhất để đánh giá hiệu suất mô hình, tỷ lệ giữa số lượng mẫu được phân loại chính xác trên cho tổng số mẫu của tập dữ liệu theo công thức sau:

|  |
| --- |
|  |

Accuracy cho biết mức độ tin cậy chung của mô hình chấm điểm. Nếu đạt giá trị cao: Mô hình dự đoán chính xác điểm tín dụng của khách hàng và giá trị thấp: Mô hình có thể đưa ra dự đoán sai lệch.

* Phương pháp Precision (độ chính xác dương) là tỷ lệ giữa số lượng mẫu được dự đoán khớp mẫu trên tổng dự đoán khớp mẫu và dự đoán đúng nhưng sai, được biểu diễn theo công thức dưới đây:

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

Precision đánh giá mức độ tin cậy của dự đoán khách hàng có điểm tín dụng cao (tốt). Nếu đạt giá trị cao: Mô hình ít dự đoán sai "tốt" cho khách hàng có điểm thấp và giá trị thấp: Mô hình có thể dự đoán sai "tốt" cho khách hàng có điểm thấp, dẫn đến việc đánh giá rủi ro sai lệch.

* Phương pháp Recall (độ nhạy) là tỷ lệ giữa số lượng mẫu được dự đoán khớp mẫu với tổng dự đoán khớp mẫu và dự đoán sai nhưng đúng được xác định theo công thức sau:

|  |
| --- |
|  |

Trong đó:

* + TP (True Positive) là tổng số trường hợp dự báo khớp mẫu đúng.
  + TN (True Negative) là tổng số trường hợp dự báo khớp với mẫu sai.
  + FP (False Positive) là tổng số trường hợp dự báo các quan sát thuộc mẫu đúng tính thành sai.
  + FN (False Negative) là tổng số trường hợp dự báo các quan sát thuộc mẫu sai tính thành đúng

Recall đánh giá khả năng xác định chính xác khách hàng có điểm tín dụng cao. Nếu đạt giá trị cao: Mô hình ít bỏ sót khách hàng có điểm cao, giúp giảm thiểu việc đánh giá thấp rủi ro và giá trị thấp: Mô hình có thể bỏ sót khách hàng có điểm cao, dẫn đến việc cấp tín dụng sai cho những người có rủi ro cao.

* Phương pháp F1 - Score là trung bình trọng số của Precision và Recall, thể hiện sự cân bằng, trong đó F1 - Score đạt giá trị tốt nhất là 1 và kém nhất là 0, chỉ số F1 - Score được xác định theo công thức sau:

|  |
| --- |
|  |

F1- Score cân bằng giữa việc dự đoán chính xác "tốt" và xác định chính xác khách hàng có điểm cao. Nết đạt giá trị cao: Mô hình có hiệu quả tốt trong việc phân biệt khách hàng có điểm tín dụng cao và thấp và giá trị thấp: Mô hình cần được cải thiện để nâng cao hiệu quả phân loại.

Các chỉ số đánh giá Accuracy, Precision, Recall, và F1-Score là những chỉ số quan trọng để đánh giá hiệu quả mô hình tín dụng. Việc sử dụng các chỉ số này giúp lựa chọn mô hình phù hợp và cải thiện hiệu quả đánh giá tín dụng.

## 4.4. Các tham số và môi trường cài đặt

*Môi trường Phần mềm:*

Ngôn Ngữ Lập Trình: Sử dụng Python trong môi trường Jupyter Notebook, một công cụ phổ biến trong khoa học dữ liệu cho phép thực hiện mã, ghi chú và hiển thị kết quả một cách tương tác.

*Thư Viện Sử Dụng:*

* pandas và numpy cho việc xử lý và phân tích dữ liệu.
* matplotlib và seaborn cho việc trực quan hóa dữ liệu.
* sklearn cho các mô hình học máy như Decision Tree, KNN, Random Forest, SVM và Adaboost, cũng như các công cụ tiền xử lý dữ liệu như MinMaxScaler.
* tensorflow.keras được sử dụng cho việc xây dựng và huấn luyện mô hình mạng nơ-ron tích chập (CNN) và mạng nơ-ron hồi quy (RNN).

*Môi trường Phần cứng:* Môi trường thử nghiệm của nhóm chúng tôi như sau: CPU Intel Core i5-8265U, RAM: 2400MHz.

*Thiết Lập Thí Nghiệm:* Được thực hiện trên tập dữ liệu đã được tiền xử lý, với việc chia tách tập dữ liệu huấn luyện và kiểm thử mô hình, và sử dụng các phương pháp đánh giá như accuracy, precision, recall, và F1-score để đánh giá hiệu quả của mô hình.

## 4.5. Các phương pháp cơ sở

Trong phần nghiên cứu này, nhóm chúng tôi thực hiện so sánh 7 mô hình để đánh giá tính hiệu quả và chính xác. Tất cả các phương pháp được thực hiện dưới cùng một điều kiện thí nghiệm để đảm bảo công bằng trong so sánh:

* Phương pháp phân lớp: Decision Tree, K-Nearest Neighbor, Random Forest, XGBoost, Naive Bayes
* Mô hình học sâu: Convolutional Neural Network, Recurrent Neural Network

Các thí nghiệm được thực hiện trên cùng một tập dữ liệu, sử dụng cùng một quy trình tiền xử lý và chia dữ liệu để đảm bảo tính khách quan. Mỗi phương pháp được đánh giá dựa trên các chỉ số như accuracy, precision, recall, và F1-score để có cái nhìn toàn diện về hiệu suất của chúng.

## 4.6. Kết quả và phân tích

*4.6.1. Các kỹ thuật học máy truyền thống*

Trong bài báo cáo, nhóm chúng tôi đã xây dựng, thử nghiệm các mô hình học máy truyền thống sau: Decision Tree (Cây quyết định), K-Nearest Neighbor (K- láng giềng gần nhất), Random Forest (Rừng ngẫu nhiên), XGBoost, Naive Bayes và có kết quả được trình bày trong bảng dưới đây:

**Bảng 3.** Kết quả đánh giá các mô hình học máy truyền thống

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Model** | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| 1 | **Decision Tree** | 0.855382 | 0.854924 | 0.855382 | 0.851688 |
| 2 | **K-Nearest Neighbor** | 0.712090 | 0.707868 | 0.712090 | 0.702121 |
| **3** | **Random Forest** | **0.885779** | **0.887773** | **0.885779** | **0.882678** |
| 4 | **XGBoost** | 0.835985 | 0.837708 | 0.835985 | 0.830886 |
| 5 | **Naive Bayes** | 0.617953 | 0.611433 | 0.617953 | 0.585627 |

Qua bảng phân tích của các mô hình học máy truyền thống khi sử dụng lần lượt các chỉ số đánh giá hiệu suất Accuracy, Precision, Recall, F1-score và kết quả cho thấy mô hình Random Forest (Rừng ngẫu nhiên) có hiệu suất đạt tốt nhất khi các chỉ số đều đạt trên 88%, cao hơn các mô hình còn lại khoảng từ 0-30% tùy vào từng phương pháp so sánh. Bên cạnh đó, Naive Bayes và K-Nearest Neighbor hai mô hình có hiệu suất được nhìn nhận là thấp nhất trong bảng phân tích với kết quả lần lượt là hơn 60% và hơn 70% ở cả bốn chỉ số đánh giá.

*4.6.2 Các kỹ thuật học sâu*

Nhóm chúng tôi xây dựng mô hình Convolutional Neural Network (Mạng neural tích chập sâu) và Recurrent Neural Network (Mạng nơ ron hồi quy), thu được kết quả sau:

**Bảng 4.** Kết quả đánh giá các mô hình học sâu

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Model** | **Accuracy** | **Precision** | **Recall** | **F1-score** | **Thời gian** |
| 1 | **CNN** | 0.686358 | 0.679902 | 0.686358 | 0.677860 | **1 phút 56s** |
| 2 | **RNN** | 0.675702 | 0.671962 | 0.675702 | 0.663117 | **10 phút 43s** |

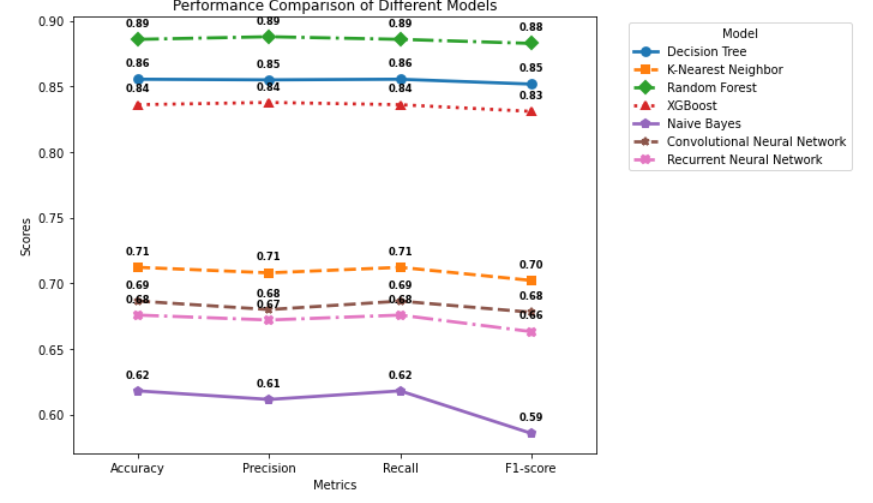
Hai mô hình ở bảng 4 đưa ra kết quả là hiệu suất ở cả 2 phương pháp đánh giá xấp xỉ nhau. Tuy nhiên, mô hình Convolutional Neural Network đã có mức % nhỉnh hơn một chút so với mô hình còn lại ví dụ như Accuracy của CNN chỉ cao hơn 1% so với RNN. Quan sát kết quả thu được có thể thấy RNN đưa ra kết quả không cao và thời gian thực nghiệm dài hơn CNN khoảng 9 phút.

*4.6.3. So sánh các mô hình*

**Bảng 5.** So sánh các mô hình

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Model** | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| 1 | **DT** | 0.855382 | 0.854924 | 0.855382 | 0.851688 |
| 2 | **KNN** | 0.712090 | 0.707868 | 0.712090 | 0.702121 |
| 3 | **RF** | **0.885779** | **0.887773** | **0.885779** | **0.882678** |
| 4 | **XGB** | 0.835985 | 0.837708 | 0.835985 | 0.830886 |
| 5 | **NB** | 0.617953 | 0.611433 | 0.617953 | 0.585627 |
| 7 | **CNN** | 0.686358 | 0.679902 | 0.686358 | 0.677860 |
| 8 | **RNN** | 0.675702 | 0.671962 | 0.675702 | 0.663117 |

Dựa vào bảng tổng hợp các chỉ số đánh giá hiệu suất của các mô hình trên, ta nhận thấy rằng mô hình Random Forest có tỷ lệ hiệu suất cao nhất trong 7 mô hình còn lại lần lượt với tất cả các chỉ số đánh giá đều trên 88%.



**Hình 7.** Đánh giá hiệu quả của các mô hình

Thông qua biểu đồ trên ta nhận thấy rằng các phương pháp đánh giá trên mô hình RF có hiệu suất cao nhất (khoảng 89%), tiếp đó là mô hình DT với hiệu suất chỉ kém 2-3%. Ngoài ra, mô hình XGB có hiệu suất cao thứ ba (khoảng 84%). Với phương pháp đánh giá hiệu suất bằng 4 chỉ số accuracy, precision, recall, f1-score thì mô hình RNN có kết quả hiệu suất là thấp nhất (trên 60%).

# V. KẾT LUẬN

Tóm lại, bài báo cáo này đã so sánh nhiều cách tiếp cận khác nhau của mô hình chấm điểm tín dụng, mô hình truyền thống với 5 mô hình cơ sở là DT, KNN, RF, XGB và NB; mô hình học sâu CNN và RNN. Việc đánh giá trên nhiều mô hình lựa chọn ra mô hình có khả năng chấm điểm tín dụng chính xác nhất. Các thực nghiệm đã chỉ ra rằng mô hình Random Forest được huấn luyện vượt trội hơn các mô hình khác với các chỉ số đánh giá hiệu năng accury, precision, recall, f1-score đều trên 88%. Đối với phương pháp xây dựng tính năng, mỗi phương pháp thể hiện thế mạnh của mình khi được đào tạo bởi các mô hình khác nhau. Tuy nhiên, cần nhấn mạnh rằng không có mô hình tính điểm tín dụng lý tưởng cũng như không có kỹ thuật tốt nhất được sử dụng để xây dựng mô hình chấm điểm tín dụng; và kỹ thuật tốt cho tập dữ liệu này chưa chắc tốt trên tập dữ liệu khác. Kỹ thuật này hơn kỹ thuật khác khả năng dự đoán trong các trường hợp cụ thể khác nhau. “Kĩ thuật tốt nhất” phụ thuộc vào từng yêu cầu cụ thể, cấu trúc của dữ liệu, các chức năng của ứng dụng, mức độ tách biệt các lớp bằng mục tiêu phân loại.

Trong công việc trong tương lai, để cải thiện hiệu suất của các mô hình được đề xuất, nhiều mô hình và kỹ thuật học máy sẽ được khám phá, chẳng hạn như Deep Forest và xếp chồng mô hình. Ngoài ra, chúng tôi đang có kế hoạch thử nghiệm các mô hình được đề xuất với nhiều bộ dữ liệu chấm điểm tín dụng hơn để xác minh tính mạnh mẽ của chúng.

**TÀI LIỆU THAM KHẢO**

1. Chacko, Annie, John Aravindhar D, and Antonidoss A. "Enhancing Credit Score Analysis: A Novel Approach with Random Forest and Kernel SVM." International Journal of Electronics and Communication Engineering (2023).
2. Yu, Lean, Rongtian Zhou, Rongda Chen, and Kin Keung Lai. "Missing Data Preprocessing in Credit Classification: One-Hot Encoding or Imputation?" Emerging Markets Finance and Trade 58 (2020): 472-482.
3. Credit score classification.https://www.kaggle.com/datasets/parisrohan/credit-score-classification/data
4. Arram, Anas, Masri Ayob, Musatafa Abbas Abbood Albadr, Alaa Sulaiman, and Dheeb Albashish. "Credit card score prediction using machine learning models: A new dataset." ArXiv (2023).
5. Abdou, Hussein A. & John Pointon. "Credit Scoring, Statistical Techniques and Evaluation Criteria: A Review of the Literature." Intell. Syst. Account. Finance Manag. 18 (2011): 59-88.
6. Marinakis, Yannis, Magdalene Marinaki, Michael Doumpos, Nikolaos F. Matsatsinis & Constantin Zopounidis. "Optimization of nearest neighbor classifiers via metaheuristic algorithms for credit risk assessment." Journal of Global Optimization 42 (2008): 279-293.
7. Baker, Mohammed Rashad, Zuhair Norii Mahmood, and Ehab Hashim Shaker. "Ensemble Learning with Supervised Machine Learning Models to Predict Credit Card Fraud Transactions." Revue d'Intelligence Artificielle (2022)
8. Wan, Xiang, Wen-Che Wang, Jiming Liu, and Tiejun Tong. "Estimating the sample mean and standard deviation from the sample size, median, range and/or interquartile range." BMC Medical Research Methodology 14 (2014)