
DỰ ĐOÁN RỦI RO TÀI CHÍNH VỚI PROFILE TUNING DỰA TRÊN PRETRAINED FOUNDATION MODELS

1stLường Minh Trí
23020440
K68-A-AI2
23020440@vnu.edu.vn

2ndKiều Đức Nam
23020404
K68-A-AI2
kieuducnam05@gmail.com

3rdNguyễn Trọng Hồng Phúc
23020410
K68-A-AI2
hongphucnguyentrong@gmail.com

4thNguyễn Quang Huy
22022582
K67-A-AI2
22022582@vnu.edu.vn

Tóm tắt nội dung

Dự đoán rủi ro tài chính đóng vai trò quan trọng trong lĩnh vực tài chính. Các phương pháp học máy đã được áp dụng rộng rãi để tự động phát hiện các rủi ro tiềm ẩn, từ đó tiết kiệm chi phí lao động. Tuy nhiên, sự phát triển trong lĩnh vực này đang bị chậm lại do hai nguyên nhân sau: 1) các thuật toán được sử dụng đang trở nên lỗi thời, đặc biệt là trong bối cảnh trí tuệ nhân tạo sinh ngôn ngữ (Generative AI) và các mô hình ngôn ngữ lớn (LLMs) đang tiến bộ nhanh chóng; 2) thiếu một bộ dữ liệu tài chính chuẩn và mã nguồn mở đã kim hãm nghiên cứu liên quan trong nhiều năm.

Để giải quyết các vấn đề này, chúng tôi đề xuất hai đóng góp: FinPT và FinBench. FinPT là một phương pháp mới trong dự đoán rủi ro tài chính bằng cách điều chỉnh hồ sơ (Profile Tuning) trên các mô hình nền đã được huấn luyện sẵn (pretrained foundation models). FinBench là tập hợp các bộ dữ liệu chất lượng cao về các loại rủi ro tài chính như vỡ nợ, gian lận, và rời bỏ khách hàng.

Trong FinPT, chúng tôi chuyển đổi dữ liệu dạng bảng thành hồ sơ khách hàng bằng ngôn ngữ tự nhiên thông qua mẫu hướng dẫn định sẵn và sử dụng LLM để tạo văn bản mô tả hồ sơ. Sau đó, các mô hình nền lớn được tinh chỉnh bằng văn bản hồ sơ này để thực hiện dự đoán. Chúng tôi chứng minh hiệu quả của FinPT thông qua thực nghiệm với các mô hình mạnh đại diện trên FinBench. Các phân tích chuyên sâu cũng góp phần nâng cao hiểu biết về việc ứng dụng LLM trong dự đoán rủi ro tài chính.

CCS Concepts: Computing methodologies → Machine learning; Natural language processing; Applied computing.

Từ khóa: Profile Tuning, Financial Risk Prediction, Financial Benchmark, Pretrained Foundation Models

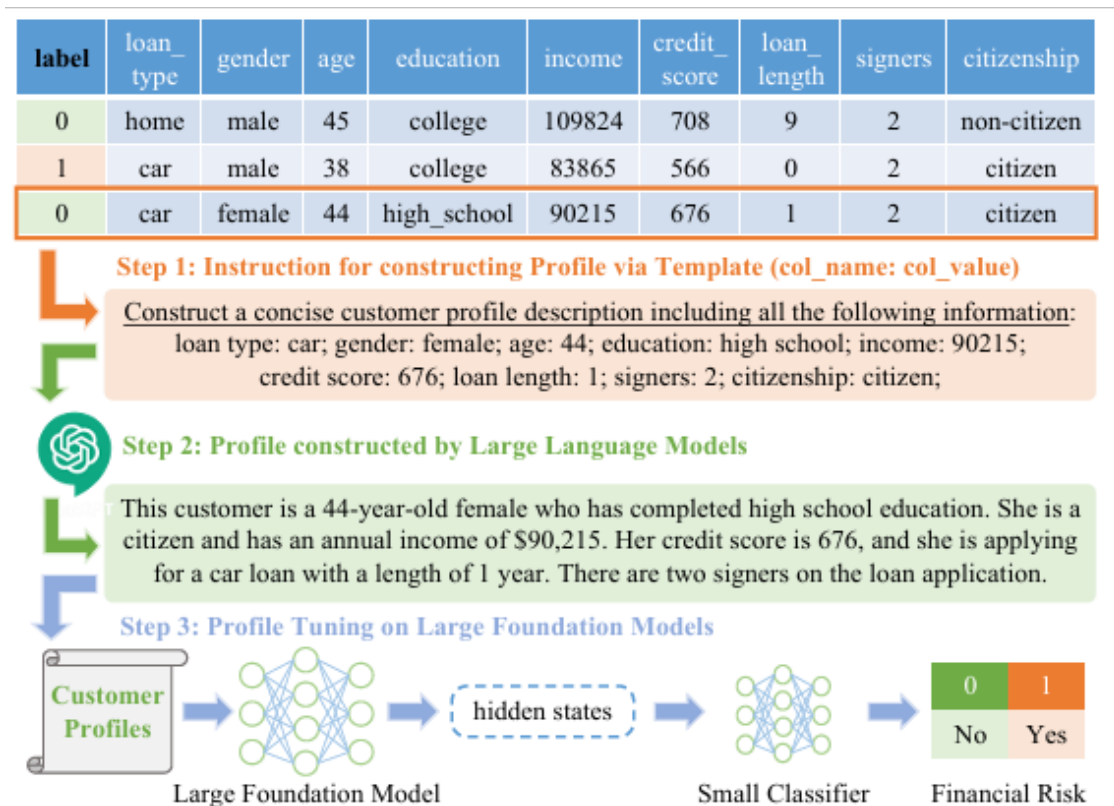
1 GIỚI THIỆU

Việc ứng dụng các phương pháp học máy đã đóng góp đáng kể trong lĩnh vực dự đoán rủi ro tài chính [12, 14, 24]. Bằng cách tận dụng các kỹ thuật này, các tổ chức tài chính có thể đánh giá tốt hơn các rủi ro tài chính liên quan đến khách hàng và đưa ra quyết định sáng suốt hơn. Quá trình tự động hóa này cũng giúp giảm thiểu lỗi do con người, từ đó nâng cao hiệu quả và độ chính xác trong việc đánh giá rủi ro tài chính.

Thông qua việc phân tích các tập dữ liệu tài chính quy mô lớn, người ta nhận thấy rằng các nhiệm vụ dự đoán rủi ro tài chính có thể được đơn giản hóa thành các bài toán phân loại hồ sơ. Những bài toán này bao gồm việc đánh giá hồ sơ khách hàng để xác định khả năng họ có thể vi phạm các quy định tài chính. Hồ sơ khách hàng thường bao gồm các thông tin cá nhân như tuổi, giới tính, trình độ học vấn và lịch sử công việc, tình trạng xã hội, cũng như các hồ sơ tài chính trước đó.

Dựa trên thực tế rằng nhiều bảng dữ liệu tài chính có tên cột tương tự nhau, chúng tôi đề xuất hợp nhất thông tin từ các bảng khác nhau và tiến hành huấn luyện mô hình quy mô lớn. Ngoài ra, được thúc đẩy bởi sự phát triển mạnh mẽ và năng lực vượt trội của các mô hình ngôn ngữ lớn (LLMs) [8, 25, 31], chúng tôi

đề xuất một phương pháp mới mang tên FinPT, thực hiện Profile Tuning với sự hỗ trợ của LLMs để dự đoán rủi ro tài chính. Tổng quan phương pháp được minh họa tại Hình 1.



Hình 1: Tổng quan phương pháp FinPT: Bước 1 điền dữ liệu bảng vào mẫu hướng dẫn, Bước 2 tạo hồ sơ ngôn ngữ tự nhiên bằng LLMs, Bước 3 tinh chỉnh mô hình nền lớn để dự đoán rủi ro.

Trái ngược với sự phát triển mạnh mẽ của các thuật toán phân loại, các tập dữ liệu tài chính hiện vẫn còn rất khan hiếm. Sự thiếu vắng một bộ dữ liệu chuẩn thống nhất đã làm chậm sự phát triển của các thuật toán dự đoán rủi ro tài chính. Xuất phát từ mối quan tâm này, chúng tôi đề xuất FinBench – một bộ dữ liệu chất lượng cao phục vụ cho nhiệm vụ dự đoán rủi ro tài chính. Cụ thể, chúng tôi thu thập hàng trăm tập dữ liệu tài chính từ nền tảng Kaggle, sau đó chọn lọc ra 10 tập dữ liệu có chất lượng cao liên quan đến ba loại rủi ro phổ biến: vỡ nợ (default), gian lận (fraud) và rời bỏ khách hàng (churn). Các tập dữ liệu này được xử lý dưới một cấu trúc dữ liệu thống nhất và cung cấp API dễ dàng để nạp vào mô hình.

FinBench chứa khoảng 333.000 mẫu đã gán nhãn, chia sẵn thành tập huấn luyện, kiểm tra và xác thực. Ngoài cặp dữ liệu X-y truyền thống dành cho các thuật toán học máy, chúng tôi còn cung cấp thông tin thống kê bổ sung về từng bảng để phục vụ các thuật toán phân loại đặc biệt. Đồng thời, với mỗi mẫu dữ liệu, chúng tôi còn cung cấp câu lệnh hướng dẫn và văn bản hồ sơ phục vụ cho phương pháp Profile Tuning.

Để đánh giá hiệu quả của FinPT, chúng tôi áp dụng Profile Tuning lên các mô hình nền mã nguồn mở như GPT-2 [29] và LLaMA [31], đồng thời so sánh phương pháp của mình với các thuật toán phân loại tiêu biểu khác, bao gồm các mô hình học máy mạnh như RandomForest [16, 19], XGBoost [9] và các mạng nơ-ron thiết kế riêng cho dữ liệu bảng như DeepFM [15] và TabNet[3]. Chúng tôi sử dụng F1-score làm thước đo hiệu suất do tất cả các tập dữ liệu đều là bài toán phân loại nhị phân. Bên cạnh đó, các mẫu dương tính (mẫu rủi ro) được áp dụng hệ số phạt cao hơn trong quá trình huấn luyện do tính mất cân bằng của dữ liệu.

Kết quả thực nghiệm trên toàn bộ tập FinBench cho thấy phương pháp của chúng tôi giúp cải thiện hiệu suất dự đoán một cách ổn định so với các phương pháp khác. Chúng tôi cũng chỉ ra rằng áp dụng Profile Tuning trên nhiều bảng khác nhau còn cho kết quả vượt trội hơn, chứng minh tính ưu việt của phương pháp so với các mô hình phân loại truyền thống vốn chỉ hoạt động trên từng bảng riêng biệt. Cuối cùng, chúng tôi cũng khám phá thêm các chiến lược tinh chỉnh LLM khác như học trong ngữ cảnh (in-context learning) và học theo hướng dẫn (instruction tuning). Kết quả cho thấy rằng, dù các LLM được gợi ý (prompted) có thể cung cấp các giải thích tài chính hữu ích, chúng vẫn không hiệu quả bằng các mô hình phân loại chuyên biệt.

2 CÁC NGHIÊN CỨU LIÊN QUAN

2.1 PHÂN LOẠI DỮ LIỆU TÀI CHÍNH

Dữ liệu tài chính thường được tổ chức dưới dạng bảng (tabular). Cụ thể, nhiều cột trong bảng là các đặc trưng (features) của khách hàng hoặc giao dịch. Phân loại dữ liệu bảng là một bài toán kinh điển trong lĩnh vực học máy [5, 23], và đã có vô số mô hình được đề xuất để giải quyết bài toán này, chẳng hạn như RandomForest [16, 19], XGBoost [9], CatBoost [28] và LightGBM [17].

Với sự bùng nổ của học sâu, nhiều mạng nơ-ron đã được phát triển nhằm xử lý dữ liệu bảng tổng quát [7], ví dụ như DeepFM [15], STG [35], VIME [40] và TabNet [3]. Trong nghiên cứu này, chúng tôi áp dụng phương pháp Profile Tuning được đề xuất – FinPT – trên dữ liệu bảng tài chính bằng cách xây dựng mô tả hồ sơ khách hàng thống nhất từ nhiều bộ dữ liệu khác nhau, từ đó mang lại một cách tiếp cận mới trong việc xử lý dữ liệu bảng trong kỷ nguyên của các mô hình ngôn ngữ lớn (LLMs).

2.2 CÁC MÔ HÌNH NỀN ĐƯỢC HUẤN LUYỆN SẴN (PRETRAINED FOUNDATION MODELS)

Trong những năm gần đây, với sự phát triển nhanh chóng của các mô hình dựa trên Transformer [32], các mô hình nền như BERT [11, 20, 22, 36], T5 [30, 33, 38, 37], và GPT [8, 29] đã chứng minh được hiệu quả trong nhiều tác vụ thông qua quy trình huấn luyện trước - tinh chỉnh sau (pretraining-finetuning).

Nhiều nghiên cứu đã đề xuất tinh chỉnh BERT trên văn bản tài chính để thực hiện các nhiệm vụ dự đoán, chẳng hạn như phân tích cảm xúc tài chính [2], phân loại cảm xúc tài chính [39], và khai phá văn bản tài chính [20].

Bên cạnh phương pháp tinh chỉnh cổ điển, các mô hình ngôn ngữ lớn gần đây [6, 8, 25, 18, 31] nổi lên với khả năng học trong ngữ cảnh (in-context learning – ICL) [13], tức là học từ một vài ví dụ trong cùng ngữ cảnh mà không cần tinh chỉnh toàn bộ mô hình.

Do đó, các nhà nghiên cứu đã đề xuất tinh chỉnh LLMs bằng cách thiết kế các chỉ dẫn (instructions) cụ thể [4, 10, 26, 27, 33, 34] để khai thác tối đa tiềm năng của các mô hình này. Trong nghiên cứu này, chúng tôi tích hợp FinPT vào các mô hình ngôn ngữ mã nguồn mở khác nhau như BERT, GPT-2 và LLaMA [31], đồng thời so sánh chiến lược tinh chỉnh của chúng tôi với hai phương pháp: học trong ngữ cảnh và học theo hướng dẫn (instruction tuning) trên tập dữ liệu FinBench.

3 PHƯƠNG PHÁP CỦA CHÚNG TÔI

Trong phần này, chúng tôi trình bày tổng quan, công thức hóa và triển khai phương pháp mới mang tên **FinPT** – một phương pháp dự đoán rủi ro tài chính bằng kỹ thuật *Profile Tuning*, tận dụng các mô hình nền lớn đã được huấn luyện trước.

3.1 TỔNG QUAN

Tổng quan về FinPT được minh họa tại Hình 1. Chiến lược của chúng tôi gồm ba bước chính như sau:

Bước 1. Chúng tôi định nghĩa một mẫu chỉ dẫn (instruction template) có dạng: “Hãy xây dựng một mô tả hồ sơ khách hàng ngắn gọn bao gồm đầy đủ các thông tin sau: TabKV”, trong đó TabKV là tập hợp các cặp “tên cột: giá trị cột;” của từng dòng dữ liệu.

Bước 2. Chúng tôi nhập chỉ dẫn từ Bước 1 vào các mô hình ngôn ngữ lớn như ChatGPT và GPT-4 [25] để tạo ra mô tả hồ sơ khách hàng mạch lạc và tự nhiên, bao gồm đầy đủ thông tin từ bảng.

Bước 3. Chúng tôi sử dụng các văn bản hồ sơ được tạo ở Bước 2 để tinh chỉnh các mô hình nền lớn như BERT [20], GPT [8, 29] và LLaMA [31]. Dựa trên các trạng thái ẩn (hidden states) từ mô hình nền, một bộ phân loại nhỏ – thường là mạng nơ-ron truyền thẳng – sẽ được sử dụng để dự đoán xem hồ sơ có rủi ro tài chính hay không.

3.2 CÔNG THỨC HOÁ

Cho một bộ dữ liệu bảng $D = \{X, \mathbf{y}, \mathbf{k}\}$, trong đó:

- $\mathbf{y} \in \{0, 1\}^m$ là nhãn nhị phân. Với $y_i = 0$ là không rủi ro, $y_i = 1$ là có rủi ro. - $X = \{\mathbf{x}_1, \dots, \mathbf{x}_m\} \in \mathbb{R}^{m \times n}$ gồm m dòng và n đặc trưng (feature). - \mathbf{k} là danh sách tên các cột.

3.2.1 Xây dựng Hồ sơ

Như đã đề cập ở Bước 1, chúng tôi chuyển dữ liệu bảng thành chỉ dẫn I_i bằng cách chèn các cặp {tên_cột : giá_trị} cho từng dòng \mathbf{x}_i . Tập hợp các chỉ dẫn là I . Sau đó, mỗi chỉ dẫn I_i được nhập vào mô hình ngôn ngữ lớn như ChatGPT thông qua API. Kết quả từ Bước 2 là tập hồ sơ văn bản tự nhiên P , chứa thông tin khách hàng được trình bày mạch lạc.

3.2.2 Profile Tuning

Ở Bước 3, chúng tôi tinh chỉnh các mô hình nền đã huấn luyện trước $F : \mathbb{R}^t \rightarrow \mathbb{R}^{t \times d}$, như BERT, GPT và LLaMA. Trong đó: - t là số lượng token đầu vào. - d là chiều của vector trạng thái ẩn.

Tập hồ sơ sau khi được mã hóa là $P \in \mathbb{R}^{m \times t} = \{\mathbf{p}_i\}_{i=1}^m$.

Do mô hình nền thường rất lớn, tinh chỉnh toàn bộ tham số là rất tốn tài nguyên. Vì vậy, chúng tôi thêm một bộ phân loại nhỏ $C : \mathbb{R}^{t \times d} \rightarrow \mathbb{R}^{d \times 2}$, là một mạng nơ-ron truyền thẳng, thực hiện phân loại nhị phân dựa trên trạng thái ẩn $\mathbf{h} = F(\mathbf{p})$.

Biểu diễn đầu ra:

$$\mathbf{h} = F(\mathbf{p}) \quad (1)$$

$$\mathbf{h}_{cls} = \begin{cases} \frac{1}{t} \sum_{i=1}^t \mathbf{h}_i, & \text{nếu dùng BERT (encoder)} \\ \mathbf{h}_t, & \text{nếu dùng GPT (decoder)} \end{cases} \quad (2)$$

$$\hat{\mathbf{y}} = C(\mathbf{h}_{cls}) \quad (3)$$

Hàm mất mát (Binary Cross Entropy): Do dữ liệu mất cân bằng, chúng tôi áp dụng học có trọng số (cost-sensitive learning). Trọng số mẫu dương (rủi ro) là:

$$w_{pos} = \frac{|\mathbf{y}| - \sum_i y_i}{\sum_i y_i}$$

Hàm mất mát BCE có trọng số:

$$\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) = -\frac{1}{m} \sum_{i=1}^m [y_i \cdot l_i \cdot w_{pos} + (1 - y_i) \cdot l_i]$$

trong đó $l_i = -y_i \log(\hat{y}_i) - (1 - y_i) \log(1 - \hat{y}_i)$.

3.2.3 Profile Tuning Trên Nhiều Bộ Dữ liệu

Cho tập gồm v bộ dữ liệu bảng:

$$D_{all} = \{D_i\}_{i=1}^v = \{X_i, \mathbf{y}_i, \mathbf{k}_i\}_{i=1}^v$$

và tập hồ sơ tương ứng:

$$P_{all} = \{P_i\}_{i=1}^v$$

Chúng tôi sử dụng tất cả các hồ sơ trong P_{all} để tinh chỉnh mô hình nền lớn F , và đánh giá hiệu quả trên từng tập kiểm tra tương ứng với mỗi D_i .

Bộ dữ liệu được thu thập từ nền tảng Kaggle¹ và được phát hành công khai tại Hugging Face².

4 Đánh giá Benchmark

4.1 Tổng quan FinBench

Bộ dữ liệu FinBench được xây dựng để phục vụ cho các tác vụ dự đoán rủi ro tài chính. Có ba loại rủi ro chính bao gồm:

- **Vỡ nợ (Default):** Bao gồm vỡ nợ thẻ tín dụng (CD) và vỡ nợ khoản vay (LD).
- **Gian lận (Fraud):** Gian lận tín dụng hoặc thanh toán không hợp lệ.
- **Rời bỏ khách hàng (Churn):** Khách hàng ngừng sử dụng dịch vụ tài chính.

Chúng tôi trình bày các thống kê của bộ FinBench trong Bảng 1, bao gồm tên tác vụ, mã tập dữ liệu, số lớp nhãn, số đặc trưng, số lượng mẫu trong tập huấn luyện, kiểm tra, xác thực và tỷ lệ mẫu dương (rủi ro) trong từng tập.

Bảng 1: **Thống kê của FinBench.** Có ba loại rủi ro tài chính chính, bao gồm: vỡ nợ, gian lận và rời bỏ khách hàng. Loại rủi ro vỡ nợ gồm hai nhóm con là vỡ nợ thẻ tín dụng (cd) và vỡ nợ khoản vay (ld). “cf” chỉ gian lận tín dụng và “cc” là rời bỏ khách hàng. Bảng thể hiện tên và mô tả tác vụ, mã tập dữ liệu, số lớp nhãn, số lượng đặc trưng, số lượng mẫu huấn luyện/kiểm tra/xác thực và tỷ lệ mẫu rủi ro (positives) trong mỗi tập.

Tác vụ	Mô tả	Dữ liệu	#Lớp	#Đặc trưng	#Huấn luyện [% Rủi ro]	#Xác thực [% Rủi ro]	#Kiểm tra [% Rủi ro]
Vỡ nợ thẻ tín dụng	Dự đoán người dùng có vỡ nợ thẻ tín dụng hay không	cd1	2	9	2738 [7.0%]	305 [6.9%]	1305 [6.2%]
		cd2	2	23	18900 [22.3%]	2100 [22.3%]	9000 [21.8%]
Vỡ nợ khoản vay	Dự đoán người dùng có vỡ nợ khoản vay hay không	ld1	2	12	2118 [8.9%]	236 [8.5%]	1010 [9.0%]
		ld2	2	11	18041 [21.7%]	2005 [20.8%]	8592 [21.8%]
		ld3	2	35	142060 [21.6%]	15785 [21.3%]	67648 [22.1%]
Gian lận tín dụng	Dự đoán người dùng có gian lận tài chính hay không	cf1	2	19	5352 [0.67%]	595 [1.1%]	2550 [0.90%]
		cf2	2	120	5418 [6.0%]	603 [7.3%]	2581 [6.0%]
Rời bỏ khách hàng	Dự đoán người dùng có rời bỏ hay không (churn)	cc1	2	9	4189 [23.5%]	466 [22.7%]	1995 [22.4%]
		cc2	2	10	6300 [20.8%]	700 [20.6%]	3000 [19.5%]
		cc3	2	21	4437 [26.1%]	493 [24.9%]	2113 [27.8%]

Các tập dữ liệu được sử dụng trong FinBench được thu thập từ các nguồn công khai trên Kaggle³, với thông tin chi tiết như sau:

- **cd1:** Credit Default Dataset by gustavotg.
- **cd2:** UCI Credit Card Default Dataset.
- **ld1:** Home Equity Loan (HMEQ) Dataset.
- **ld2:** Credit Risk Dataset.
- **ld3:** Vehicle Loan Default Dataset.
- **cf1:** Credit Card Fraud Detection Dataset.
- **cf2:** Credit Card Usage Dataset.

Toàn bộ bộ dữ liệu FinBench được công bố tại Hugging Face: yuweiyin/FinBench.

¹<https://www.kaggle.com/>

²<https://huggingface.co/datasets/yuweiyin/FinBench>

³<https://www.kaggle.com/>

4.2 Rủi ro loại 1: Vỡ nợ (Default)

Vỡ nợ được định nghĩa là việc không thể hoàn trả gốc hoặc lãi đúng hạn cho khoản nợ đang có. Hiện tượng này có thể xảy ra với cá nhân, doanh nghiệp hoặc thậm chí quốc gia, và là một yếu tố đáng quan ngại đối với các chủ nợ. Trong FinBench, rủi ro vỡ nợ được chia thành hai nhóm con:

- **Vỡ nợ thẻ tín dụng (CD):** Khi người dùng không thể thanh toán đúng hạn hóa đơn thẻ tín dụng.
- **Vỡ nợ khoản vay (LD):** Khi người dùng không thể trả nợ định kỳ với các khoản vay như vay thế chấp, vay thuê, hoặc vay mua xe.

Mục tiêu là dự đoán liệu một người dùng sẽ vỡ nợ hay không. Các tập dữ liệu trong nhóm này bao gồm:

- **cd1:** Bao gồm dữ liệu từ hàng nghìn khách hàng tại một tổ chức tài chính, chứa các thông tin như mã định danh, trạng thái tín dụng, loại khoản vay, giới tính, độ tuổi, trình độ học vấn, thu nhập, điểm tín dụng và các thông tin liên quan.
- **cd2:** Ghi lại thông tin khách hàng sử dụng thẻ tín dụng tại Đài Loan từ tháng 4 đến tháng 9 năm 2005, bao gồm tình trạng thanh toán, nhân khẩu học, thông tin tín dụng, lịch sử thanh toán và sao kê thẻ.
- **ld1:** Dữ liệu vay vốn HMEQ (Home Equity), gồm các thông tin chi tiết về khoản vay thế chấp, cùng biến mục tiêu nhị phân thể hiện khách hàng có vỡ nợ hay không.
- **ld2:** Gồm hàng chục nghìn bản ghi khoản vay với các đặc trưng như độ tuổi, thu nhập hàng năm, thời gian làm việc, loại hình nhà ở, mục đích vay, số tiền vay, lãi suất, lịch sử tín dụng và tỷ lệ vỡ nợ trước đó.
- **ld3:** Gồm hơn 200 nghìn bản ghi về vay mua xe, bao gồm thông tin về khoản vay, người vay và lịch sử vay mượn.

4.3 Rủi ro loại 2: Gian lận (Fraud)

Gian lận tài chính là hành vi cố ý giả mạo thông tin hoặc danh tính nhằm đánh lừa hệ thống hoặc người khác, như sử dụng trái phép thẻ tín dụng/thẻ ghi nợ, hoặc gửi thông tin sai lệch để trục lợi tài sản.

Trong FinBench, có hai tập dữ liệu liên quan đến gian lận tín dụng:

- **cf1:** Tập hợp hành vi sử dụng thẻ tín dụng của nhiều người dùng trong vòng 6 tháng, làm cơ sở phát hiện gian lận.
- **cf2:** Gồm lịch sử sử dụng thẻ tín dụng với 120 đặc trưng khác nhau phục vụ cho việc phát hiện hành vi bất thường hoặc gian lận.

Mục tiêu là dự đoán liệu người dùng có thực hiện hành vi gian lận hay không.

4.4 Rủi ro loại 3: Rời bỏ khách hàng (Churn)

Rời bỏ khách hàng (Customer churn) đề cập đến tỷ lệ khách hàng ngừng sử dụng sản phẩm hoặc dịch vụ của doanh nghiệp trong một khoảng thời gian nhất định. Chỉ số này cung cấp thông tin quan trọng về nhóm khách hàng hiện tại có khả năng sẽ không tiếp tục mua hàng trong tương lai. Việc giảm tỷ lệ rời bỏ khách hàng là một mục tiêu then chốt đối với nhiều doanh nghiệp.

Dự đoán rời bỏ khách hàng (hoặc "attrition") không chỉ giúp giữ chân khách hàng mà còn là cơ hội để tối ưu hóa doanh thu. Tỷ lệ rời bỏ cao ảnh hưởng trực tiếp đến chi phí doanh nghiệp, làm giảm doanh thu và tăng chi phí marketing để thu hút khách hàng mới.

Trong FinBench, ba tập dữ liệu sau được sử dụng để đánh giá rủi ro churn:

- **cc1:** Gồm thông tin nhân khẩu học và lịch sử hoạt động của khách hàng ngân hàng.
- **cc2:** Cũng là dữ liệu về rời bỏ khách hàng ngân hàng, bao gồm các đặc trưng như điểm tín dụng, vị trí địa lý, giới tính, độ tuổi, thời gian gắn bó, số dư tài khoản,...
- **cc3:** Mô phỏng thông tin của một công ty viễn thông cung cấp dịch vụ điện thoại và internet tại nhà cho hàng nghìn khách hàng. Dữ liệu chỉ rõ khách hàng nào đã rời bỏ, ở lại, hoặc đăng ký dịch vụ.

5 Thiết lập Thực nghiệm

Trong phần này, chúng tôi trình bày chi tiết toàn bộ các thiết lập thực nghiệm, bao gồm: các phương pháp so sánh cơ sở (Mục 5.1), mô hình nền được sử dụng cho FinPT (Mục 5.2), chi tiết triển khai (Mục 5.3), chi tiết huấn luyện (Mục 5.4), và tiêu chí đánh giá (Mục 5.5).

5.1 Các phương pháp cơ sở (Baselines)

Để đánh giá hiệu quả của FinPT, chúng tôi so sánh nó với một loạt các phương pháp cơ sở mạnh, bao gồm các mô hình cây quyết định và mạng nơ-ron sâu được thiết kế riêng cho dữ liệu dạng bảng.

Các mô hình tăng cường dựa trên cây (Tree-based Gradient Boosting Models). Chúng tôi sử dụng các mô hình RandomForest [16, 19], XGBoost [9], CatBoost [28] và LightGBM [17] làm baseline vì hiệu suất vượt trội của chúng trong nhiều bài toán khoa học dữ liệu. Đây đều là các mô hình học tổ hợp (ensemble learning) hoặc thuật toán tăng cường gradient (gradient boosting) dựa trên cây quyết định.

Mạng nơ-ron sâu (Deep Neural Networks). Chúng tôi chọn bốn mô hình mạng nơ-ron được thiết kế để xử lý dữ liệu bảng như sau:

- **DeepFM** [15]: Kỹ thuật phổ biến trong công nghiệp, kết hợp giữa máy phân rã ma trận (factorization machines) và học sâu để học đặc trưng thông qua kiến trúc mạng nơ-ron mới.
- **STG** [35]: Một khung huấn luyện giúp học đồng thời hàm hồi quy hoặc phân loại phi tuyến và lựa chọn đặc trưng bằng cách sử dụng cơ chế cổng ngẫu nhiên (Stochastic Gates).
- **VIME** [40]: Khung học tự giám sát và bán giám sát cho dữ liệu bảng, sử dụng kỹ thuật điền giá trị (Value Imputation) và ước lượng mặt nạ (Mask Estimation).
- **TabNet** [3]: Kiến trúc học sâu có khả năng giải thích cao, sử dụng cơ chế chú ý tuần tự (sequential attention) để chọn các đặc trưng quan trọng nhằm phục vụ suy luận trong dữ liệu bảng.

5.2 Mô hình nền tăng được huấn luyện sẵn như kiến trúc trục chính(Backbone)

Chúng tôi áp dụng FinPT lên các mô hình nền đã được huấn luyện trước với vai trò là kiến trúc trục chính (backbone) của mô hình.

Bảng 2 thể hiện thiết lập chi tiết của các mô hình nền được sử dụng để huấn luyện FinPT. “#Params-All” là tổng số tham số trong mô hình nền, còn “#Params-T” là số tham số có thể huấn luyện (không tính bộ phân loại nhỏ, vốn chỉ chứa dưới 1 triệu tham số).

Bảng 2: Thiết lập thực nghiệm của các mô hình nền trong FinPT.

Mô hình	#Params-All	#Params-T	Các module được huấn luyện
BERT-Base [11]	110M	110M	Toàn bộ mô hình
FinBERT [39]	110M	110M	Toàn bộ mô hình
GPT-2 [29]	117M	117M	Toàn bộ mô hình
T5-Base [30]	220M	220M	Toàn bộ mô hình
Flan-T5-Base [10]	220M	220M	Toàn bộ mô hình
T5-XXL [30]	11B	268M	Khối Decoder cuối cùng
Flan-T5-XXL [10]	11B	260M	Khối Decoder cuối cùng
LLaMA-7B [31]	7B	202M	Lớp Decoder cuối cùng
LLaMA-13B [31]	13B	317M	Lớp Decoder cuối cùng

Tóm tắt các mô hình nền:

- **BERT** [11]: Mô hình Transformer hai chiều, chỉ có encoder, được huấn luyện với hai nhiệm vụ: mặt nạ từ (Masked Language Modeling) và dự đoán câu tiếp theo.

- **FinBERT [39]**: Mô hình BERT được tinh chỉnh trên văn bản tài chính để phục vụ các tác vụ phân loại như phân tích cảm xúc.
- **GPT-2 [29]**: Mô hình decoder-only dùng trong sinh ngôn ngữ, được huấn luyện theo dạng dự đoán từ tiếp theo (autoregressive).
- **T5 [30]**: Mô hình encoder-decoder chuyển tất cả tác vụ NLP về định dạng text-to-text thống nhất.
- **Flan-T5 [10]**: Phiên bản T5 được tinh chỉnh với hướng dẫn (instruction tuning), hỗ trợ nhiều tác vụ zero-shot và few-shot.
- **LLaMA [31]**: Bộ mô hình nền mã nguồn mở, huấn luyện trên hàng nghìn tỷ token, với các phiên bản từ 7B đến 65B tham số. Đáng chú ý, LLaMA-13B vượt trội hơn GPT-3 (175B) [8] trên nhiều benchmark.

Như đã đề cập trong Mục 3.2, do giới hạn tài nguyên tính toán, chúng tôi chỉ thêm một bộ phân loại nhỏ (feed-forward neural network) để học từ các trạng thái ẩn (hidden states) của mô hình nền. Đồng thời, các mô-đun trong mô hình nền được đóng băng một phần tùy theo kích thước mô hình, như trình bày trong Bảng 2.

Nguồn dữ liệu:

- **cc1**: Jobathon March 2022 - Bank Churn Dataset.
- **cc2**: Churn Prediction for Bank Customers.
- **cc3**: Telco Customer Churn Dataset (IBM).

5.3 Chi tiết Triển khai (Implementation Details)

Chúng tôi triển khai các mô hình cơ sở sử dụng mã nguồn chính thức được công bố hoặc thư viện chuẩn. Cụ thể:

- Với mô hình RandomForest, chúng tôi sử dụng hiện thực từ thư viện `scikit-learn`⁴.
- Đối với XGBoost, sử dụng `XGBClassifier`⁵.
- Đối với CatBoost, sử dụng `CatBoostClassifier`⁶.
- Với LightGBM, sử dụng `LGBMClassifier`⁷.

Với các mô hình mạng nơ-ron, chúng tôi tái hiện lại DeepFM, STG, VIME và TabNet dựa trên mã nguồn mở từ dự án TabSurvey⁸.

Về bước 2 trong pipeline chính: Mỗi chỉ dẫn được gửi đến ChatGPT thông qua OpenAI Python API với đoạn mã sau:

```
import openai
response = openai.ChatCompletion.create(
    model="gpt-3.5-turbo",
    messages=[
        {"role": "system", "content": "You are a helpful financial assistant."},
        {"role": "user", "content": instruction[i]},
    ],
    temperature=0,
)
```

Danh sách các chỉ dẫn `instruction[i]` được xây dựng từ bước 1 như đã mô tả trong Mục 3.2.1.

⁴<https://scikit-learn.org/>

⁵<https://xgboost.readthedocs.io/>

⁶<https://catboost.ai/>

⁷<https://lightgbm.readthedocs.io/>

⁸<https://github.com/Understand-AI/TabSurvey>

Về bước 3: Chúng tôi tải mô hình nền lớn từ nền tảng Hugging Face⁹ cùng với tokenizer tương ứng. Các mã mô hình tương ứng với Bảng 2 như sau:

- bert-base-cased
- yiyanghkust/finbert-pretrain
- gpt2
- t5-base
- google/flan-t5-base
- t5-11b
- google/flan-t5-xxl
- openlm-research/open_llama_7b
- openlm-research/open_llama_13b

5.4 Chi tiết Huấn luyện (Training Details)

Toàn bộ thí nghiệm được thực hiện trên hai GPU NVIDIA A40, mỗi GPU có bộ nhớ 48GB. Chúng tôi sử dụng chế độ BF16 do kiến trúc Ampere hỗ trợ, cho phép huấn luyện mô hình nền lớn bằng phép toán độ chính xác hỗn hợp (mixed precision). Cụ thể, các phép toán được thực hiện chủ yếu ở độ chính xác nửa (FP16) để tăng tốc độ huấn luyện, trong khi vẫn lưu trữ thông tin quan trọng ở độ chính xác đơn (FP32) để đảm bảo độ ổn định của mạng.

Đối với tất cả các mô hình cơ sở (baseline), chúng tôi thiết lập batch size là 128 và số epoch tối đa là 100.

Đối với phương pháp Profile Tuning trong FinPT:

- Batch size tối đa được thiết lập là 128.
- Độ dài chuỗi tối đa (sequence length) là 128. Nếu chuỗi đầu vào dài hơn, độ dài sẽ được nâng lên 256 và batch size giảm một nửa.
- Ký tự đệm (padding token) được đặt giống với ký tự kết thúc câu (end-of-sentence token) trong tokenizer.
- Learning rate được đặt là 5×10^{-5} và hệ số suy giảm trọng số (weight decay) là 0.01.
- Bộ tối ưu hóa sử dụng là AdamW [21].

FinBench cung cấp sẵn các tập huấn luyện, xác thực và kiểm tra. Trong quá trình huấn luyện, mô hình sử dụng tập huấn luyện để cập nhật tham số, và đánh giá trên tập xác thực để chọn checkpoint tốt nhất. Sau huấn luyện, checkpoint tốt nhất được nạp lại để đánh giá trên tập kiểm tra.

Mỗi thí nghiệm được lặp lại 4 lần với các seed ngẫu nhiên $\in \{0, 1, 42, 1234\}$ và chúng tôi báo cáo kết quả trung bình của các lần chạy.

5.5 Chỉ số đánh giá (Evaluation Metrics)

Trong tất cả các thí nghiệm trên FinBench, chúng tôi sử dụng F1-score [1] làm thước đo chính để đánh giá hiệu suất mô hình, vì tất cả các tập dữ liệu trong FinBench đều là bài toán phân loại nhị phân mất cân bằng.

So với độ chính xác (Accuracy), F1-score phù hợp hơn trong trường hợp này vì nó giảm thiểu tình trạng dương giả (false negatives), vốn dễ xảy ra khi mẫu dương chiếm tỷ lệ thấp.

6 Kết quả và Phân tích (Results and Analysis)

Trong phần này, chúng tôi trình bày và phân tích toàn bộ kết quả thực nghiệm theo thiết lập đã mô tả trong Mục 5.

⁹<https://huggingface.co/>

6.1 Kết quả chính (Main Results)

Chúng tôi báo cáo điểm F1-score cho tác vụ dự đoán rủi ro tài chính trên bộ FinBench tại Bảng 3. Trung bình, FinPT vượt trội hơn các phương pháp dựa trên cây quyết định và các mô hình mạng nơ-ron trước đây với khoảng cách đáng kể, đặc biệt khi áp dụng Profile Tuning đầy đủ lên các mô hình nền lớn như GPT-2, T5 và Flan-T5.

Phân tích các mô hình dựa trên cây: Trung bình, CatBoost đạt hiệu suất tốt nhất trong số bốn thuật toán tăng cường dựa trên cây trên FinBench, mặc dù trong một số tập dữ liệu, Random Forest, XGBoost hoặc LightGBM lại vượt trội hơn. Những mô hình này vốn là baseline mạnh trong nhiều cuộc thi phân loại, nên việc đạt F1-score trung bình trong khoảng 44–47 cho thấy FinBench là một bộ dữ liệu đầy thách thức và vẫn còn nhiều không gian để cải thiện mô hình.

Phân tích các mô hình mạng nơ-ron: Trong các mạng nơ-ron cho dữ liệu bảng, TabNet là mô hình hiệu quả nhất trên 8 trong số 10 tập dữ liệu, trong khi DeepFM đạt điểm cao nhất ở 2 tập còn lại. Tuy nhiên, nhóm mô hình này vẫn chưa vượt qua được các mô hình cây và FinPT, cho thấy rằng mạng nơ-ron với quy mô nhỏ, dù có kiến trúc thiết kế chuyên biệt cho dữ liệu bảng, vẫn chưa đủ mạnh để dự đoán rủi ro hiệu quả.

Phân tích FinPT (huấn luyện toàn bộ – Tuning All): FinPT khi áp dụng lên các mô hình nền khác nhau cho thấy hiệu suất dự đoán không giống nhau. Khi tinh chỉnh toàn bộ tham số, Flan-T5 là backbone tốt nhất cho FinPT, đạt kết quả tốt nhất trên 8 tập dữ liệu. Điều này cho thấy sức mạnh vượt trội của Flan-T5 trong dự đoán rủi ro tài chính. Việc Flan-T5 liên tục vượt T5 chứng minh tầm quan trọng của tinh chỉnh theo hướng dẫn (instruction tuning) và mở rộng quy mô mô hình.

Chúng tôi cũng nhận thấy rằng GPT-2 liên tục vượt trội hơn các mô hình BERT với khoảng cách đáng kể, cho thấy Transformer dạng decoder-only sử dụng trạng thái cuối (last-token hidden state) có thể là bộ phân loại tốt hơn so với Transformer encoder-only sử dụng trạng thái lớp cuối. Ngoài ra, BERT và FinBERT có hiệu suất tương đương, điều này cho thấy việc huấn luyện mô hình nền trên văn bản tài chính chưa chắc đã mang lại lợi ích nếu dữ liệu tiền huấn luyện và dữ liệu tinh chỉnh thuộc các miền khác nhau.

Phân tích FinPT (chỉ tinh chỉnh lớp cuối – Tuning Last): Khi chỉ tinh chỉnh lớp Decoder cuối cùng trong các mô hình nền, chúng tôi nhận thấy LLaMA luôn cho hiệu suất vượt trội hơn ba mô hình còn lại trên FinBench. Đặc biệt, LLaMA-13B vượt trội hơn LLaMA-7B, phù hợp với các phát hiện trước đó từ nghiên cứu LLaMA [31]. Đáng chú ý, chỉ với lớp cuối cùng có thể huấn luyện, LLaMA-13B vẫn đạt hiệu suất rất cao, cho thấy rõ hiệu quả của phương pháp FinPT.

6.2 Profile Tuning trên toàn bộ tập dữ liệu

Do chúng tôi chuyển đổi các tập dữ liệu bảng thành văn bản hồ sơ theo cấu trúc thống nhất, việc thực hiện Profile Tuning đồng thời trên tất cả các tập là khả thi.

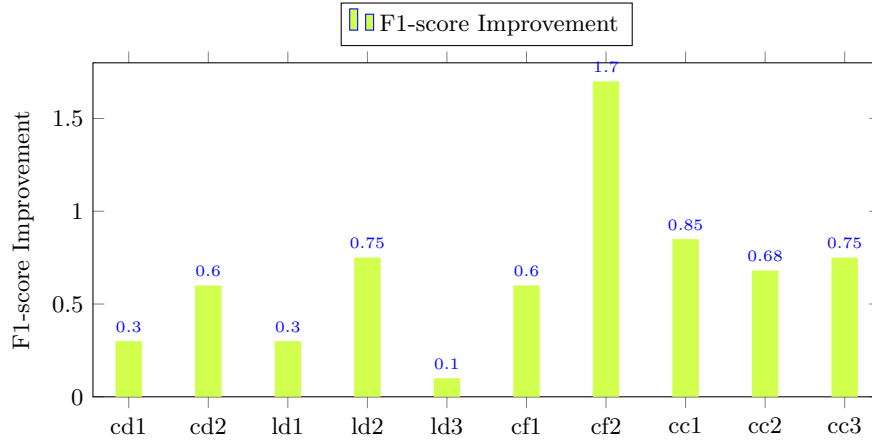
Như minh họa ở Hình 2, chúng tôi thực hiện thí nghiệm này với mô hình hiệu quả nhất trong Bảng 3, tức FinPT với Flan-T5-Base. Kết quả cho thấy huấn luyện trên toàn bộ tập FinBench thay vì từng tập riêng lẻ giúp cải thiện hiệu suất một cách ổn định trên hầu hết các bộ kiểm tra.

Tập dữ liệu **cf2** hưởng lợi nhiều nhất từ việc huấn luyện hợp nhất, do ban đầu có điểm số thấp nhất. Điều này chứng minh rằng việc tận dụng dữ liệu chéo tập có thể giúp tăng khả năng tổng quát và độ chính xác của mô hình.

Bảng 3: **Kết quả dự đoán rủi ro tài chính trên FinBench.** Chúng tôi báo cáo điểm F1-score cho 10 tập dữ liệu trong FinBench và điểm trung bình (Avg). Trong cột “Huấn luyện”, “Grid Search” nghĩa là sử dụng tìm kiếm lưới siêu tham số cho các mô hình cây, “From Scratch” nghĩa là huấn luyện từ đầu, “Tune All” là tinh chỉnh toàn bộ mô hình nền, và “Tune Last” là chỉ tinh chỉnh lớp cuối. Giá trị tốt nhất mỗi hàng được in đậm.

Model Class	Model	Training	CD1	CD2	LD1	LD2	LD3	CF1	CF2	CC1	CC2	CC3	Avg
Tree-based	RandomForest [16, 19]	Grid Search	23.0	52.2	47.9	64.2	38.9	40.8	19.2	41.8	52.1	62.8	44.29
	XGBoost [9]	Grid Search	22.7	45.5	56.3	76.4	40.4	46.7	20.5	40.1	57.1	61.1	46.68
	CatBoost [28]	Grid Search	21.8	52.2	48.3	72.8	41.4	46.2	21.0	41.0	59.3	65.6	46.91
	LightGBM [17]	Grid Search	21.7	52.6	45.2	71.4	40.8	40.0	21.0	41.0	59.3	65.5	45.95
NN for Table	DeepFM [15]	From Scratch	8.4	39.8	46.7	78.0	15.3	43.2	8.0	11.9	55.3	59.3	36.59
	STG [35]	From Scratch	7.1	40.9	23.5	53.9	10.2	20.2	5.2	5.8	37.3	38.6	24.27
	VIME [40]	From Scratch	8.9	41.8	37.5	75.3	18.2	41.7	7.4	20.4	53.2	56.3	36.07
	TabNet [3]	From Scratch	10.1	44.5	40.6	77.5	24.2	45.7	9.7	23.1	57.2	59.9	39.25
FinPT Tune All	BERT-Base [11]	Tune All	19.2	50.4	47.1	79.6	42.1	45.5	5.6	27.9	60.2	64.1	44.17
	FinBERT [39]	Tune All	18.3	50.8	45.9	80.9	41.9	45.1	5.9	28.2	60.1	64.5	44.16
	GPT-2 [29]	Tune All	23.0	52.5	49.4	81.7	43.3	47.4	8.6	37.2	60.7	66.1	46.99
	T5-Base [30]	Tune All	23.4	53.1	48.3	81.4	45.2	49.2	11.7	42.1	61.3	67.1	48.28
	Flan-T5-Base [10]	Tune All	23.8	53.3	48.9	82.8	45.8	49.5	13.5	43.7	61.9	68.5	49.17
FinPT Tune Last	T5-XXL [30]	Tune Last	21.9	49.8	44.7	73.3	40.2	42.6	6.1	38.2	58.7	63.4	43.89
	Flan-T5-XXL [10]	Tune Last	22.4	50.1	45.1	75.1	40.7	42.9	6.0	38.9	59.2	63.8	44.42
	LLaMA-7B [31]	Tune Last	22.7	51.6	46.4	76.7	41.8	44.2	8.4	40.4	60.1	64.6	45.69
	LLaMA-13B [31]	Tune Last	22.9	52.0	47.2	79.2	42.4	45.7	9.2	41.8	60.4	65.2	46.60

Lưu ý: Điểm F1 trên tập dữ liệu **ld3** cải thiện ít nhất, chủ yếu vì đây là tập dữ liệu lớn nhất và chiếm tỷ trọng lớn trong toàn bộ FinBench.



Hình 2: Biểu đồ cải thiện F1-score của FinPT (Flan-T5-Base) khi được huấn luyện trên toàn bộ tập dữ liệu kết hợp so với huấn luyện riêng lẻ từng tập.

6.3 Các Chiến Lược Khác Nhau

Như đã đề cập trong Mục 2, *học trong ngữ cảnh* (In-context Learning - ICL) và *tinh chỉnh theo hướng dẫn* (Instruction Tuning - IT) là hai xu hướng phổ biến trong việc khai thác các mô hình ngôn ngữ lớn (LLMs).

Đối với ICL, chúng tôi đưa vào mô hình GPT-2 [29], Flan-T5 [10] và LLaMA [31] năm ví dụ huấn luyện, với định dạng:

Profile: \mathbf{p}_i \$
Answer: \mathbf{y}_i \$ (với $i \in \{1, 2, 3, 4, 5\}$)

19. <https://chat.openai.com/>
20. <https://huggingface.co/models>
21. F1-score in scikit-learn

Câu truy vấn cuối cùng có dạng:

```
Profile: $\mathbf{p}_t$ $  
Option 1: Yes. Option 2: No.  
Answer:
```

Trong đó \mathbf{p}_t là hồ sơ cần dự đoán, và mô hình sẽ xuất ra nhãn ở dạng văn bản “Yes” hoặc “No”.

Đối với IT, chúng tôi sử dụng một câu lệnh hướng dẫn trực tiếp:

```
Predict whether the following customer is financially risky. $\mathbf{p}_t$ $  
Option 1: Yes. Option 2: No.  
Answer:
```

Chúng tôi cũng thử nghiệm với nhiều câu lệnh hướng dẫn chi tiết khác.

Do mô hình nền thường tạo ra câu trả lời mơ hồ hoặc tránh kết luận rõ ràng, chúng tôi xem một dự đoán là đúng nếu đầu ra chứa chính xác từ khóa nhãn “Yes” hoặc “No”. Mặc dù đây là tiêu chí đánh giá khá thoáng, nhưng cả ICL và IT đều đạt F1-score rất thấp (dưới 10), kém xa so với các mô hình phân loại truyền thống.

Tuy vậy, chúng tôi nhận thấy rằng các đầu ra này vẫn hữu ích ở khía cạnh hỗ trợ giải thích cho con người. Ví dụ dưới đây minh hoạ điều đó:

“Thật khó để dự đoán chắc chắn liệu khách hàng này có rủi ro tài chính hay không dựa trên thông tin hạn chế. Tuy nhiên, dựa trên dữ liệu đã cung cấp, có vẻ như khách hàng này không có rủi ro cao. Cô ấy có số dư lớn, thu nhập tốt, đã gắn bó với ngân hàng một thời gian, trạng thái giao dịch là 1.0. Ngoài ra, loại tín dụng ở mức trung bình — không tốt nhưng cũng không tệ. Do đó, Lựa chọn 2: Không, cô ấy không rủi ro tài chính, là câu trả lời hợp lý hơn.”

7 Kết luận

Trong nghiên cứu này, chúng tôi đã đề xuất **FinPT** — một phương pháp mới nhằm chuyển đổi dữ liệu bảng tài chính thành các hồ sơ khách hàng dưới dạng văn bản, sau đó áp dụng kỹ thuật *Profile Tuning* trên các mô hình nền quy mô lớn (large foundation models) để thực hiện dự đoán rủi ro.

Bên cạnh đó, chúng tôi cũng giới thiệu **FinBench** — một bộ benchmark dành cho bài toán dự đoán rủi ro tài chính, bao gồm các tập dữ liệu chất lượng cao liên quan đến ba loại rủi ro phổ biến: *vỡ nợ* (default), *gian lận* (fraud), và *rời bỏ khách hàng* (churn).

Hiệu quả của phương pháp FinPT đã được xác thực thông qua việc đánh giá với nhiều mô hình cơ sở mạnh trên bộ FinBench. Ngoài ra, các phân tích chuyên sâu còn giúp mang lại cái nhìn rõ hơn về tiềm năng ứng dụng của các mô hình ngôn ngữ nền quy mô lớn trong lĩnh vực dự đoán rủi ro tài chính.

Mặc dù FinPT cho thấy hiệu quả vượt trội so với các phương pháp truyền thống, nghiên cứu này vẫn tồn tại một số hạn chế. Thứ nhất, quá trình *Profile Tuning* yêu cầu tài nguyên tính toán lớn khi áp dụng với các mô hình nền quy mô rất lớn (như LLaMA-13B hoặc T5-XXL). Thứ hai, việc biểu diễn hồ sơ khách hàng dưới dạng ngôn ngữ tự nhiên vẫn chưa hoàn toàn tối ưu hoá cho mọi loại bảng dữ liệu, đặc biệt là các tập dữ liệu có cấu trúc phức tạp hoặc chứa thông tin thiếu hụt.

Trong tương lai, chúng tôi kỳ vọng sẽ mở rộng FinPT theo ba hướng chính. Thứ nhất, tích hợp thêm các phương pháp nén mô hình (model compression) và tinh chỉnh tham số hiệu quả (efficient fine-tuning) để giảm chi phí huấn luyện. Thứ hai, cải tiến kỹ thuật xây dựng hồ sơ bằng cách kết hợp thêm tri thức tài chính từ nguồn bên ngoài (external knowledge). Cuối cùng, chúng tôi cũng mong muốn khám phá khả năng áp dụng FinPT trong các lĩnh vực phi tài chính như y tế, thương mại điện tử, và tuyển dụng — nơi dữ liệu bảng và dự đoán rủi ro cũng đóng vai trò thiết yếu.

8 TÀI LIỆU THAM KHẢO

Tài liệu

- [1] F1-score in sklearn. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html. Accessed: 2025-05-29.
- [2] Dogu Araci. Finbert: Financial sentiment analysis with pre-trained language models. *CoRR*, abs/1908.10063, 2019.
- [3] Sercan O Arik and Tomas Pfister. Tabnet: Attentive interpretable tabular learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6679–6687. AAAI Press, 2021.
- [4] Jiaqi Bai, Zhao Yan, Jian Yang, Xinnian Liang, Hongcheng Guo, and Zhoujun Li. Knowprefix-tuning: A two-stage prefix-tuning framework for knowledge-grounded dialogue generation. *CoRR*, abs/2306.15430, 2023.
- [5] Christopher M Bishop. *Pattern recognition and machine learning*. Springer, New York, NY, USA, 5th edition, 2007.
- [6] Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, et al. Gpt-neox-20b: An open-source autoregressive language model. *CoRR*, abs/2204.06745, 2022.
- [7] Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. Deep neural networks and tabular data: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 1(1):1–21, 2022.
- [8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, 2020.
- [9] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, San Francisco, CA, USA, 2016. ACM.
- [10] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416, 2022.
- [11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT 2019*, pages 4171–4186, Minneapolis, MN, USA, 2019. Association for Computational Linguistics.
- [12] Matthew F Dixon, Igor Halperin, and Paul Bilokon. *Machine learning in finance*, volume 1170. Springer, New York, NY, USA, 2020.
- [13] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. A survey for in-context learning. *CoRR*, abs/2301.00234, 2023.
- [14] John W Goodell, Satish Kumar, Weng Marc Lim, and Debidutta Pattnaik. Artificial intelligence and machine learning in finance: Identifying foundations, themes, and research clusters from bibliometric analysis. *Journal of Behavioral and Experimental Finance*, 32:100577, 2021.
- [15] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. Deepfm: A factorization-machine based neural network for ctr prediction. In *Proceedings of IJCAI*, pages 1725–1731, Melbourne, Australia, 2017. ijcai.org.
- [16] Tin Kam Ho. Random decision forests. In *Third International Conference on Document Analysis and Recognition, ICDAR 1995*, volume 1, pages 278–282, Montreal, Canada, 1995. IEEE Computer Society.
- [17] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30, pages 3146–3154, 2017.
- [18] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilic, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. Bloom: A 176b parameter open-access multilingual language model. *CoRR*, abs/2211.05100, 2022.
- [19] Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.

-
- [20] Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. Finbert: A pre-trained financial language representation model for financial text mining. In *IJCAI*, pages 4513–4519, Yokohama, Japan, 2020. ijcai.org.
 - [21] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, New Orleans, LA, USA, 2019. OpenReview.net.
 - [22] Shuming Ma, Jian Yang, Haoyang Huang, Zewen Chi, Li Dong, Dongdong Zhang, Hany Hassan Awadalla, Alexandre Muzio, Akiko Eriguchi, Saksham Singhal, et al. Xlm-t: Scaling up multilingual machine translation with pretrained cross-lingual transformer encoders. *CoRR*, abs/2012.15547, 2020.
 - [23] Ryszard Stanislaw Michalski, Jaime Guillermo Carbonell, and Tom M Mitchell. *Machine learning: An artificial intelligence approach*. Springer Science & Business Media, New York, NY, USA, 2013.
 - [24] Noella Nazareth and Yeruva Venkata Ramana Reddy. Financial applications of machine learning: A literature review. *Expert Systems with Applications*, 219:119640, 2023.
 - [25] OpenAI. Gpt-4 technical report. *CoRR*, abs/2303.08774, 2023.
 - [26] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744, New Orleans, LA, USA, 2022.
 - [27] Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4. *CoRR*, abs/2304.03277, 2023.
 - [28] Liudmila Ostroumova Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. In *Advances in Neural Information Processing Systems*, volume 31, pages 6639–6649, 2018.
 - [29] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. OpenAI blog, 2019. 1(8), 9.
 - [30] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1):5485–5551, 2020.
 - [31] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971, 2023.
 - [32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008, Long Beach, CA, USA, 2017.
 - [33] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations (ICLR)*. OpenReview.net, 2022. Virtual Event.
 - [34] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837, 2022.
 - [35] Yutaro Yamada, Ofir Lindenbaum, Sahand Negahban, and Yuval Kluger. Feature selection using stochastic gates. In *International Conference on Machine Learning*, volume 119, pages 10648–10659. PMLR, 2020.
 - [36] Jian Yang, Shuming Ma, Dongdong Zhang, Shuangzhi Wu, Zhoujun Li, and Ming Zhou. Alternating language modeling for cross-lingual pre-training. In *AAAI Conference on Artificial Intelligence*, pages 9386–9393, New York, NY, USA, 2020. AAAI Press.
 - [37] Jian Yang, Shuming Ma, Haoyang Huang, Dongdong Zhang, Li Dong, Shaohan Huang, Alexandre Muzio, Saksham Singhal, Hany Hassan, Xia Song, and Furu Wei. Multilingual machine translation systems from microsoft for wmt21 shared task. In *WMT@EMNLP*, pages 446–455. Association for Computational Linguistics, 2021.
 - [38] Jian Yang, Shuming Ma, Li Dong, Shaohan Huang, Haoyang Huang, Yuwei Yin, Dongdong Zhang, Liqun Yang, Furu Wei, and Zhoujun Li. Ganlm: Encoder-decoder pre-training with an auxiliary discriminator. In *ACL*, pages 9394–9412, Toronto, Canada, 2023. Association for Computational Linguistics.

-
- [39] Yi Yang, Mark Christopher Siy Uy, and Allen Huang. Finbert: A pretrained language model for financial communications. *CoRR*, abs/2006.08097, 2020.
 - [40] Jinsung Yoon, Yao Zhang, James Jordon, and Mihaela van der Schaar. Vime: Extending the success of self-and semi-supervised learning to tabular domain. In *Advances in Neural Information Processing Systems*, volume 33, pages 11033–11043, 2020.