FINPT

Group 10

Financial Risk Prediction with Profile Tuning on Pretrained Foundation Models

GROUP MEMBERS

Lường Minh Trí
23020440

Nguyễn Trọng Hồng Phúc
23020410

Kiều Đức Nam
23020404

Nguyễn Quang Huy
22022582

OVERALL

01 Abstract

02 Introduction

03 Related Work

04 Method

05 Experiment SetUp

06 Result and analyst

07 Conclusion

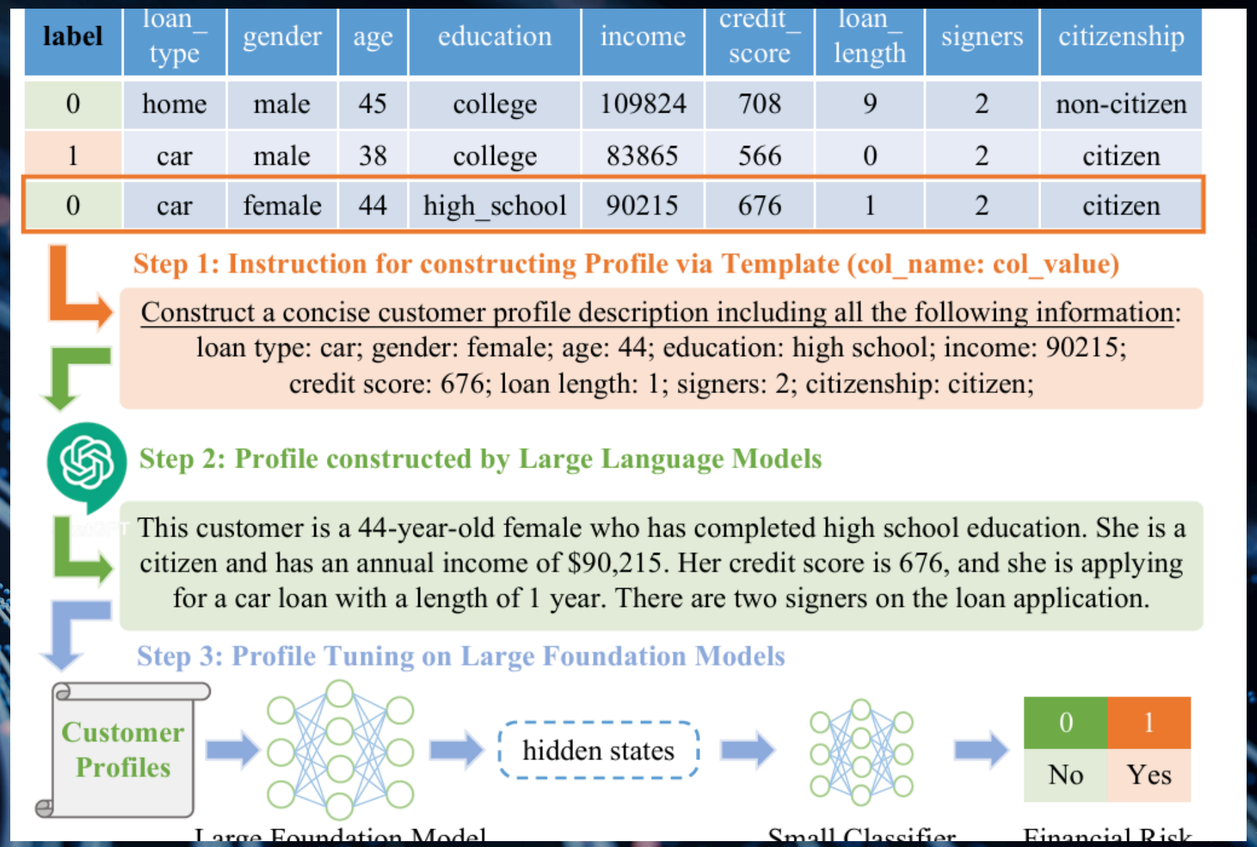
1. ABSTRACT

- 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 và do đó tiết kiệm chi phí lao động.

- Hiện nay, các phương pháp trên dần trở nên lạc hậu vì:

1) Các thuật toán được sử dụng đã lỗi thời, đặc biệt là trong bối cảnh AI tạo sinh và các mô hình ngôn ngữ lớn (LLM) phát triển nhanh chóng;
2) việc thiếu một chuẩn mực tài chính thống nhất và nguồn mở đã cản trở các nghiên cứu liên quan trong nhiều năm.

2. INTRODUCTION



label	loan_type	gender	age	education	income	credit_score	loan_length	signers	citizenship
0	home	male	45	college	109824	708	9	2	non-citizen
1	car	male	38	college	83865	566	0	2	citizen
0	car	female	44	high_school	90215	676	1	2	citizen

- Sử dụng phương pháp FinPT và FinBench.
- FinPT: Phương pháp sử dụng Large Language Models (LLMs) để chuyển đổi dữ liệu dạng bảng thành mô tả hồ sơ tự nhiên.
- Quy trình:
  - Lấp đầy mẫu hướng dẫn với dữ liệu bảng.
  - Sử dụng LLMs (ChatGPT) để tạo mô tả hồ sơ tự nhiên.
  - Tinh chỉnh các mô hình nền tảng với mô tả hồ sơ và bộ phân loại nhỏ để dự đoán rủi ro tài chính.

2. INTRODUCTION

- FinBench: Bộ dữ liệu tiêu chuẩn cao gồm 333K mẫu có nhãn từ các rủi ro tài chính như vỡ nợ, lừa đảo, và rời bỏ dịch vụ. (Giải quyết vấn đề thiếu hụt data tài chính)
- Chức năng:
  - Cung cấp cấu trúc dữ liệu thống nhất.
  - Bao gồm dữ liệu huấn luyện, kiểm tra, và phân chia cho mô hình.
  - Thêm mô tả hồ sơ và hướng dẫn cho quá trình Profile Tuning.

FINBENCH – BỘ DỮ LIỆU CHUẨN ĐÁNH GIÁ RỦI RO TÀI CHÍNH BẰNG AI

FinBench – Công cụ để chuẩn hóa dữ liệu, chuẩn hóa đánh giá, tối ưu mô hình dự đoán rủi ro tài chính. Là một bộ benchmark toàn diện giúp đánh giá mô hình AI trong ba bài toán tài chính thực tế: vỡ nợ, gian lận và rời bỏ dịch vụ.

FINTECH

Nội dung:

- Bộ dữ liệu chuẩn hóa dùng để đánh giá các mô hình học máy trong lĩnh vực tài chính.
- Kết hợp dữ liệu bảng và hồ sơ văn bản.
- Tập hợp từ hàng trăm bộ dữ liệu trên Kaggle, chọn lọc ra 10 bộ chất lượng cao.

bản của bạn

CẤU TRÚC VÀ PHÂN CHIA DỮ LIỆU

Vậy dữ liệu FinBench có gì?

Với hơn 333.000 mẫu dữ liệu

Cấu trúc dữ liệu của FinBench

FinBench bao gồm tổng cộng khoảng 333.000 mẫu dữ liệu được gán nhãn. Mỗi tập dữ liệu được chia thành ba phần:

- Tập huấn luyện (train): 70% tổng dữ liệu
- Tập kiểm định (validation): 10% tổng dữ liệu
- Tập kiểm tra (test): 30% tổng dữ liệu

Mỗi tập dữ liệu có nhãn phân loại thành 2 nhóm:

- 1: Có rủi ro tài chính
- 0: Không có rủi ro tài chính

Đặc trưng của bài toán tài chính:

- Là sự mất cân bằng dữ liệu
- Số lượng đặc trưng có sự khác biệt

MỤC TIÊU VÀ PHẠM VI CỦA FINBENCH

FinBench giúp gì cho mô hình học máy?

Tập trung vào 3 loại rủi ro tài chính thực tế:

Vỡ nợ (Default)

Khả năng một khách hàng không thể trả nợ đúng hạn.

- Vỡ nợ thẻ tín dụng
- Vỡ nợ khoản vay

Gian lận (Fraud)

Gian lận tài chính là hành vi cố ý làm sai lệch thông tin để trục lợi, bao gồm gian lận thẻ tín dụng, đánh cắp danh tính, giao dịch tài chính bất hợp pháp,...

Rời bỏ dịch vụ (Churn)

Rời bỏ khách hàng là tỷ lệ khách hàng ngừng sử dụng sản phẩm hoặc dịch vụ của công ty trong một khoảng thời gian nhất định.

- Dự đoán rủi ro rời bỏ giúp công ty giảm chi phí tiếp thị và giữ chân khách hàng hiệu quả hơn.

QUY ƯỚC

1. Rủi ro vỡ nợ (Default)

Rủi ro vỡ nợ xảy ra khi một cá nhân hoặc tổ chức không thể hoàn thành các khoản thanh toán nợ đúng hạn. FinBench chia rủi ro vỡ nợ thành hai nhóm nhỏ:

- Credit Default (CD): Khách hàng không thể thanh toán hóa đơn thẻ tín dụng đúng hạn.
- Loan Default (LD): Khách hàng không thể trả nợ vay đúng hạn (như vay thế chấp, vay mua xe, v.v.).

Các tập dữ liệu:

- cd1: Dữ liệu về khách hàng của một tổ chức tài chính, bao gồm thông tin cá nhân và lịch sử tín dụng.
- cd2: Thông tin khách hàng thẻ tín dụng tại Đài Loan từ tháng 4 đến tháng 9 năm 2005.
- ld1: Dữ liệu vay thế chấp nhà (Home Equity dataset - HMEQ).
- ld2: Dữ liệu hàng chục nghìn khoản vay với thông tin về độ tuổi, thu nhập, mức độ tín nhiệm, v.v.
- ld3: Dữ liệu hơn 200.000 hồ sơ vay mua xe.

2. Rủi ro gian lận (Fraud)

Gian lận tài chính là hành vi sử dụng thông tin sai lệch để trục lợi tài chính. FinBench cung cấp hai bộ dữ liệu về gian lận thẻ tín dụng:

- cf1: Dữ liệu hành vi sử dụng thẻ tín dụng của nhiều khách hàng trong 6 tháng.
- cf2: Dữ liệu lịch sử giao dịch thẻ tín dụng với 120 đặc trưng khác nhau.

3. Rủi ro rời bỏ dịch vụ (Churn)

Rủi ro rời bỏ dịch vụ là tỷ lệ khách hàng ngừng sử dụng dịch vụ tài chính, ảnh hưởng đến doanh thu và chi phí marketing của doanh nghiệp. FinBench cung cấp ba bộ dữ liệu về churn:

- cc1: Dữ liệu về nhân khẩu học và hoạt động của khách hàng ngân hàng.
- cc2: Dữ liệu về churn khách hàng ngân hàng với thông tin tín dụng, độ tuổi, giới tính, điểm tín dụng, v.v.
- cc3: Dữ liệu về churn của khách hàng trong một công ty viễn thông giả định.

BẢNG THỐNG KÊ TẬP DỮ LIỆU TRONG FINBENCHESSMENT

Nhóm rủi ro	Tập dữ liệu	Số lớp	Số đặc trưng	Tập huấn luyện (Pos%)	Tập kiểm định (Pos%)	Tập kiểm tra (Pos%)
Vỡ nợ thẻ tín dụng (Credit Default)	cd1	2	9	2,738 (7.0%)	305 (6.9%)	1,305 (6.2%)
	cd2	2	23	18,900 (22.3%)	2,100 (22.3%)	9,000 (21.8%)
Vỡ nợ khoản vay (Loan Default)	ld1	2	12	2,118 (8.9%)	236 (8.5%)	1,010 (9.0%)
	ld2	2	11	18,041 (21.7%)	2,005 (20.8%)	8,592 (21.8%)
	ld3	2	35	142,060 (21.6%)	15,785 (21.3%)	67,648 (22.1%)
Gian lận tài chính (Credit Fraud)	cf1	2	19	5,352 (0.67%)	595 (1.1%)	2,550 (0.90%)
	cf2	2	120	5,418 (6.0%)	603 (7.3%)	2,581 (6.0%)
Rời bỏ dịch vụ (Customer Churn)	cc1	2	9	4,189 (23.5%)	466 (22.7%)	1,995 (22.4%)
	cc2	2	10	6,300 (20.8%)	700 (20.6%)	3,000 (19.5%)
	cc3	2	21	4,437 (26.1%)	493 (24.9%)	2,113 (27.8%)

1. CREDIT DEFAULT – VỠ NỢ THẺ TÍN DỤNG (CD1, CD2)

Vỡ nợ thẻ tín dụng (Credit Default)	cd1	2	9	2,738 (7.0%)	305 (6.9%)	1,305 (6.2%)
	cd2	2	23	18,900 (22.3%)	2,100 (22.3%)	9,000 (21.8%)

- cd1 có dữ liệu nhỏ hơn và tỷ lệ vỡ nợ thấp → bài toán mất cân bằng nghiêm trọng
- cd2 có lượng dữ liệu lớn và tỷ lệ vỡ nợ cao hơn, phù hợp cho huấn luyện

2. LOAN DEFAULT – VỠ NỢ KHOẢN VAY (LD1, LD2, LD3)

Vỡ nợ khoản vay (Loan Default)	ld1	2	12	2,118 (8.9%)	236 (8.5%)	1,010 (9.0%)
	ld2	2	11	18,041 (21.7%)	2,005 (20.8%)	8,592 (21.8%)
	ld3	2	35	142,060 (21.6%)	15,785 (21.3%)	67,648 (22.1%)

- ld3 là tập lớn nhất trong toàn bộ benchmark → lý tưởng để thử các mô hình phức tạp (transformer, deep tabular...)
- ld1 có quy mô nhỏ và cũng bị mất cân bằng nhẹ

3. CREDIT FRAUD – GIAN LẬN TÀI CHÍNH (CF1, CF2)

Gian lận tài chính (Credit Fraud)	cf1	2	19	5,352 (0.67%)	595 (1.1%)	2,550 (0.90%)
	cf2	2	120	5,418 (6.0%)	603 (7.3%)	2,581 (6.0%)

- cf1 là tập mất cân bằng nặng nhất (dữ liệu gian lận chỉ chiếm ~0.67%)
- Đây là bài toán rất khó, cần dùng kỹ thuật oversampling hoặc mô hình phát hiện bất thường (anomaly detection)

4. CUSTOMER CHURN – RỜI BỎ DỊCH VỤ (CC1, CC2, CC3)

Rời bỏ dịch vụ (Customer Churn)	cc1	2	9	4,189 (23.5%)	466 (22.7%)	1,995 (22.4%)
	cc2	2	10	6,300 (20.8%)	700 (20.6%)	3,000 (19.5%)
	cc3	2	21	4,437 (26.1%)	493 (24.9%)	2,113 (27.8%)

- Các tập churn có tỷ lệ rủi ro trung bình (~20–27%), không quá mất cân bằng
- cc3 có tỷ lệ churn cao nhất, cc2 gần với phân bố thực tế

TRỞ NGẠI TRONG BỘ DỮ LIỆU

1. Mất cân bằng dữ liệu (Imbalance)

a. Biểu hiện

Nhiều tập có Pos% rất thấp:

- cf1: 0.67%
- cd1: 7.0%
- ld1: 8.9%

b. Ảnh hưởng

-Mô hình thiên về nhãn 0, bỏ qua nhãn 1 (rủi ro thực sự)

c. Giải pháp

- Oversampling (SMOTE, ADASYN)
- Undersampling
- Loss function có trọng số (class-weighted)
- Focal Loss

2. Số lượng mẫu không đồng đều

a. Biểu hiện

cf1 chỉ có 5352 mẫu train

ld3 có tới 142.060 mẫu train

b. Ảnh hưởng

- Một số tập quá nhỏ → dễ overfit
- Một số tập quá lớn → tốn tài nguyênc. Giải pháp

c. Giải pháp

- Dùng kỹ thuật regularization
- Với tập nhỏ: dùng mô hình đơn giản hơn hoặc cross-validation

3. Nhiều đặc trưng, nhưng không rõ vai trò

a. Biểu hiện

-cf1 chỉ có 5352 mẫu train

-cf2 có đến 120 đặc trưng

-cd1 chỉ có 9 đặc trưng

b. Ảnh hưởng

-Gây nhiễu, làm mô hình khó học nếu không chọn lọc kỹ

c. Giải pháp

- Feature selection (PCA, SHAP, Permutation importance)
- Loại bỏ cột không liên quan

TỔNG KẾT & KẾT LUẬN

- Bộ dữ liệu FinBench cung cấp nhiều tập con phục vụ cho các tác vụ tài chính thực tế như:
  - dự đoán vỡ nợ, gian lận tín dụng, rời bỏ khách hàng,... nhưng cũng đi kèm với nhiều thách thức quan trọng trong khai thác dữ liệu.
- Các vấn đề nổi bật bao gồm:
  - Mất cân bằng dữ liệu nghiêm trọng, đặc biệt ở bài toán phát hiện gian lận (tỉ lệ dương tính rất thấp).
  - Số lượng đặc trưng và mẫu không đồng đều giữa các tập → yêu cầu linh hoạt trong thiết kế mô hình.
  - Dữ liệu phức tạp, phân phối lệch và có outlier → đòi hỏi xử lý trước khi huấn luyện.
  - Khó đánh giá mô hình nếu chỉ dựa vào Accuracy → cần dùng thêm các chỉ số như F1, ROC-AUC, PR-AUC.

→ Việc nhận diện rõ những trở ngại này là bước khởi đầu bắt buộc để xây dựng pipeline phân tích và mô hình hóa hiệu quả và tin cậy.

- Đồng thời, chúng cũng phản ánh rõ thách thức thực tế trong lĩnh vực tài chính – nơi dữ liệu thường phân bố không đều, nhiễu, và khó đoán định, đòi hỏi những chiến lược học máy cẩn trọng và bài bản

2. INTRODUCTION

Profile Tuning

1. Fill pre-defined instruction template with tabular data in each row
2. instruct LLM like ChatGPT → Generate natural- language profile descriptions containing all information in each table row
3. Use the profile text → fine-tune foundation models with small classifier for making predictions

label	loan_type	gender	age	education	income	credit_score	loan_length	signers	citizenship
0	home	male	45	college	109824	708	9	2	non-citizen
1	car	male	38	college	83865	566	0	2	citizen
0	car	female	44	high_school	90215	676	1	2	citizen

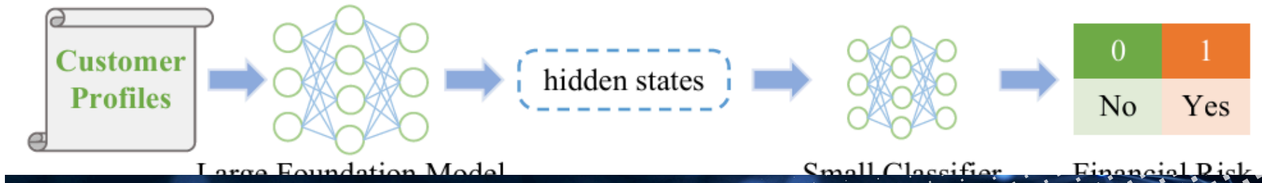
Step 1: Instruction for constructing Profile via Template (col_name: col_value)

Construct a concise customer profile description including all the following information: loan type: car; gender: female; age: 44; education: high school; income: 90215; credit score: 676; loan length: 1; signers: 2; citizenship: citizen;

Step 2: Profile constructed by Large Language Models

This customer is a 44-year-old female who has completed high school education. She is a citizen and has an annual income of $90,215. Her credit score is 676, and she is applying for a car loan with a length of 1 year. There are two signers on the loan application.

Step 3: Profile Tuning on Large Foundation Models



2. INTRODUCTION

Evaluate Effectiveness

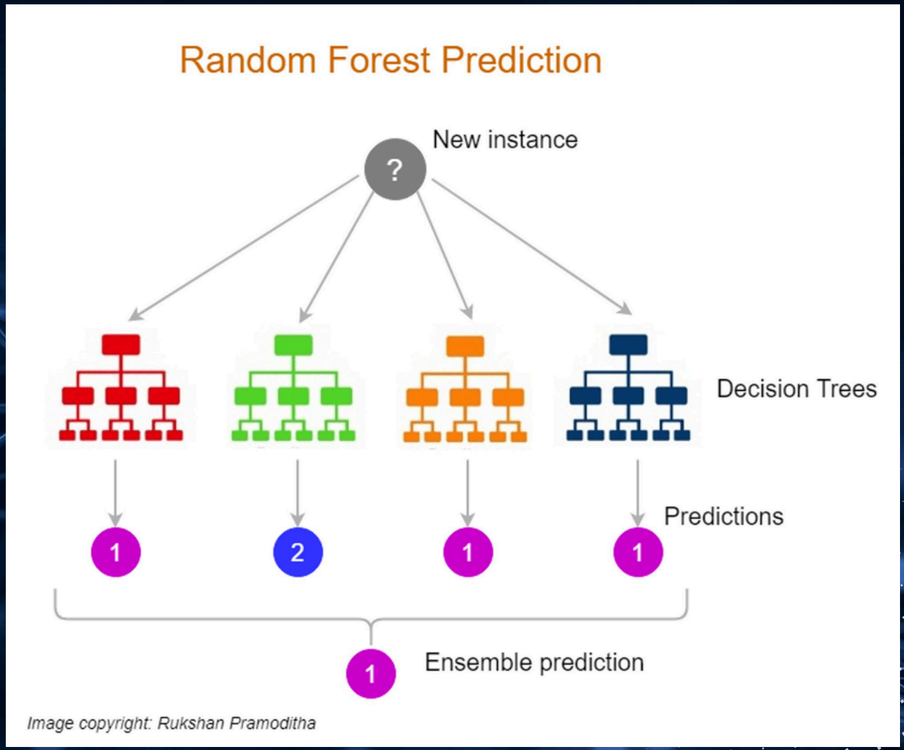
Đánh giá độ hiệu quả của FinPT

→ Apply Profile Tuning trên các open-sourced model GPT-2 và LLaMA

→ Compare FinPT vs classification algorithms, including strong machine learning baselines such as RandomForest and XGBoost, specially designed neural networks such as DeepFM and TabNet

GPT-2

LLaMA



2. INTRODUCTION

Evaluate Effectiveness

Sử dụng F1-score để đánh giá model performance vì tất cả datasets là binary classification. Ngoài ra, positive samples (risky instances) có loss penalty cao hơn khi huấn luyện vì tính không cân bằng của datasets.

→ Train models để nhạy cảm hơn nhằm phát hiện các thực thể có nguy cơ (like fraud, default, or churn), mặc dù có ít trong bộ dữ liệu.

→ Đảm bảo model không chỉ dự đoán những thứ chủ yếu mà còn thực sự hiệu quả trong việc nhận dạng và xử lí những trường hợp hiếm gặp, nhưng quan trọng đáng để tâm.

3. RELATED WORK

3.1 PHÂN LOẠI DỮ LIỆU TÀI CHÍNH (FINANCIAL DATA CLASSIFICATION):

- Bản chất của dữ liệu tài chính:
  - Dữ liệu tài chính thường ở dạng bảng (tabular data), trong đó mỗi hàng thể hiện một khách hàng hoặc một giao dịch, còn mỗi cột là một đặc trưng (feature) như tuổi, thu nhập, điểm tín dụng, số lần trễ hạn....
  - Bài toán phân loại dữ liệu tài chính rất phổ biến, ví dụ: dự đoán khả năng vỡ nợ, phát hiện gian lận, hoặc xác định khách hàng rời bỏ dịch vụ.

3. RELATED WORK

3.1 PHÂN LOẠI DỮ LIỆU TÀI CHÍNH (FINANCIAL DATA CLASSIFICATION):

- Phương pháp truyền thống, nhiều mô hình máy học đã được áp dụng cho bài toán này, điển hình là:
  - RandomForest, XGBoost, CatBoost, LightGBM: Những mô hình này nổi bật với khả năng xử lý dữ liệu bảng hiệu quả, đặc biệt trong việc xử lý các bài toán phân loại nhị phân (binary classification).
  - Hạn chế: Các mô hình này chỉ xem xét dữ liệu dạng bảng thuần túy, không tận dụng được các mối quan hệ tiềm ẩn giữa các đặc điểm khi đặt chúng trong ngữ cảnh cụ thể.

3. RELATED WORK

3.1 PHÂN LOẠI DỮ LIỆU TÀI CHÍNH (FINANCIAL DATA CLASSIFICATION):

- Sự ra đời của mô hình học sâu:
- Với sự phát triển của học sâu (deep learning), nhiều mạng nơ-ron được thiết kế riêng cho dữ liệu dạng bảng, bao gồm:
  - DeepFM: Kết hợp giữa Factorization Machines và mạng nơ-ron sâu để học mối quan hệ giữa các đặc trưng.
  - STG (Sparse Tabular Gradient): Một mô hình giúp chọn lọc đặc trưng quan trọng trong dữ liệu bảng.
  - VIME (Variational Information Maximizing Embedding): Sử dụng phương pháp học bán giám sát (semi-supervised learning) để cải thiện độ chính xác trong dự đoán.
  - TabNet: Tích hợp cơ chế attention, cho phép mô hình tập trung vào các đặc trưng quan trọng trong từng bước ra quyết định.

Trong công trình này sẽ sử dụng phương pháp Profile Tuning được đề xuất FinPT trên dữ liệu bảng tài chính bằng cách xây dựng các mô tả hồ sơ khách hàng thống nhất trên các tập dữ liệu tài chính khác nhau, cung cấp một cách mới để xử lý dữ liệu bảng trong thời đại của các mô hình ngôn ngữ lớn (LLM).

3. RELATED WORK

3.1 PHÂN LOẠI DỮ LIỆU TÀI CHÍNH (FINANCIAL DATA CLASSIFICATION):

- Điểm đổi mới của FinPT:
  - Phương pháp FinPT (Financial Profile Tuning) được đề xuất trong bài báo giúp khắc phục những hạn chế của các mô hình truyền thống bằng cách:
    - Chuyển đổi dữ liệu dạng bảng thành mô tả hồ sơ khách hàng bằng ngôn ngữ tự nhiên.
    - Tận dụng sức mạnh của các mô hình ngôn ngữ lớn (LLMs) để phân tích hồ sơ tài chính trong bối cảnh rủi ro.

→ Đây là một cách tiếp cận mới mẻ, giúp xử lý dữ liệu bảng trong kỷ nguyên của LLMs.

3. RELATED WORK
3.2 PRETRAINED FOUNDATION MODELS

- Sự phát triển của các mô hình Transformer:
  - Các mô hình Transformer như BERT, T5, GPT đã chứng minh tính hiệu quả vượt trội trong nhiều tác vụ xử lý ngôn ngữ tự nhiên (NLP).

- Quy trình tiền huấn luyện và tinh chỉnh (Pretraining-Finetuning):
  - Các mô hình này được huấn luyện trước trên lượng lớn văn bản (pretraining) rồi tinh chỉnh lại (finetuning) cho các tác vụ cụ thể như: phân tích cảm xúc tài chính, phân loại tin tức tài chính, hay khai thác văn bản tài chính.
  - Hạn chế: Việc áp dụng trực tiếp những mô hình này vào bài toán tài chính gặp khó khăn vì dữ liệu tài chính chủ yếu ở dạng bảng, không phải văn bản.

3. RELATED WORK
3.2 PRETRAINED FOUNDATION MODELS

- Sự đổi mới của FinPT, nghiên cứu này tích hợp phương pháp FinPT vào các mô hình LLMs mã nguồn mở như:
  - BERT: Chuyên xử lý các tác vụ cần hiểu mối quan hệ giữa các từ trong một câu.
  - GPT-2: Giỏi trong việc sinh văn bản và dự đoán các chuỗi văn bản tiếp theo.
  - LLaMA: Một mô hình ngôn ngữ lớn tối ưu hóa cho các tác vụ xử lý ngôn ngữ với chi phí tính toán thấp hơn.

- Đồng thời, nhóm tác giả cũng so sánh chiến lược FinPT với In-context Learning và Instruction Tuning trên tập dữ liệu FinBench, nhằm khai phá tiềm năng của LLMs trong lĩnh vực dự đoán rủi ro tài chính.

3. RELATED WORK
3.2 PRETRAINED FOUNDATION MODELS

- Các chiến lược tinh chỉnh LLMs:
  - In-context Learning (ICL):
    - Cho phép mô hình học từ một vài ví dụ điển hình trong ngữ cảnh trước khi thực hiện dự đoán, thay vì phải huấn luyện lại toàn bộ mô hình.
  - Instruction Tuning:
    - Cung cấp cho mô hình các hướng dẫn chi tiết (instructions) để điều chỉnh hành vi của mô hình trong các tác vụ cụ thể, giúp cải thiện độ chính xác khi áp dụng vào các bài toán tài chính.

4. METHOD

4.1 OVERVIEW

- Tổng quan về FinPT được minh họa trong Hình 1. Chiến lược của chúng tôi gồm ba bước chính như sau:
  - Bước 1: Định nghĩa một mẫu hướng dẫn dưới dạng "Hãy tạo một mô tả khách hàng ngắn gọn bao gồm tất cả các thông tin sau: TabKV", trong đó TabKV là tất cả các cặp "tên_cột: giá_trị;" của mỗi hàng trong bảng.
  - Bước 2: Đưa hướng dẫn từ Bước 1 vào các mô hình ngôn ngữ lớn như ChatGPT hoặc GPT-4 để tạo ra mô tả khách hàng trôi chảy và nhất quán, bao gồm tất cả thông tin tabular từ từng hàng dữ liệu.
  - Bước 3: Sử dụng các hồ sơ mô tả được tạo trong Bước 2 để tinh chỉnh các mô hình nền tảng lớn như BERT, GPT, và LLaMA. Một bộ phân loại nhỏ—thường là mạng nơ-ron truyền thẳng—được sử dụng để dự đoán liệu hồ sơ có mang rủi ro tài chính hay không.

4. METHOD

4.2 FORMULATION

- Cho một tập dữ liệu bảng D = {X, $\boldsymbol{y}$, $\boldsymbol{k}$}, trong đó $\boldsymbol{y} \in \{0, 1\}^{m}$ là các nhãn nhị phân. $\boldsymbol{y}_i$ = 0 nghĩa là mẫu i không rủi ro, còn $\boldsymbol{y}_i$ = 1 nghĩa là mẫu này có rủi ro tài chính. X = {$\boldsymbol{x}_1$, ..., $\boldsymbol{x}_m$} $\in \mathbb{R}^{m \times n}$ chứa m mẫu dữ liệu, mỗi mẫu $\boldsymbol{x}_i \in \mathbb{R}^{n}$ có n đặc trưng. $\boldsymbol{k}$ là tập hợp các tên cột trong bảng. {X, $\boldsymbol{y}$} đủ để các thuật toán máy học xử lý.

4.2.1 PROFILE CONSTRUCTION

Chuyển đổi dữ liệu bảng thành một hướng dẫn I_i bằng cách điền TabKV với {$\boldsymbol{k}_j : \boldsymbol{x}_{ji}$} $_{nj=1}$ cho mẫu i. Sau đó, chúng tôi sử dụng API OpenAI để nhập I_i vào mô hình ngôn ngữ lớn ChatGPT và tạo ra tập hồ sơ P.

4. METHOD

4.2 FORMULATION

4.2.2 PROFILE TUNING

- Trong Bước 3, chúng tôi tinh chỉnh các mô hình nền tảng đã được huấn luyện trước $F: R^{t} \to R^{t \times d}$, như BERT, GPT, và LLaMA, với các hồ sơ khách hàng, trong đó t là số lượng token đầu vào và d là số chiều của trạng thái ẩn. Chúng tôi sử dụng bộ mã hóa (tokenizer) chính thức của từng mô hình nền tảng.
- Tập hồ sơ sau khi được mã hóa có dạng:
  - $P \in R^{m \times t} = \{\boldsymbol{p}_i\}_{i=1}^{m}$
- Vì một số mô hình quá lớn, việc tinh chỉnh tất cả tham số sẽ tốn nhiều tài nguyên. Do đó, chúng tôi thêm một bộ phân loại nhỏ $C: R^{t \times d} \to R^{d \times u}$ vào cuối mô hình nền tảng, trong đó $u = 2$ là số lớp nhãn. Bộ phân loại là một mạng nơ-ron truyền thẳng nhỏ để thực hiện phân loại nhị phân dựa trên trạng thái ẩn $\boldsymbol{h} \in R^{t \times d}$ được tạo bởi mô hình nền tảng:
- Với mô hình BERT, trạng thái ẩn được tính bằng cách lấy trung bình tất cả các token chưa bị che:

4. METHOD

4.2 FORMULATION

4.2.2 PROFILE TUNING

- Trong Bước 3, chúng tôi tinh chỉnh các mô hình nền tảng đã được huấn luyện trước $F: R^{t} \to R^{t \times d}$, như BERT, GPT, và LLaMA, với các hồ sơ khách hàng, trong đó t là số lượng token đầu vào và d là số chiều của trạng thái ẩn. Chúng tôi sử dụng bộ mã hóa (tokenizer) chính thức của từng mô hình nền tảng.
- Tập hồ sơ sau khi được mã hóa có dạng:
  - $P \in R^{m \times t} = \{\boldsymbol{p}_i\}_{i=1}^{m}$
- Vì một số mô hình quá lớn, việc tinh chỉnh tất cả tham số sẽ tốn nhiều tài nguyên. Do đó, chúng tôi thêm một bộ phân loại nhỏ $C: R^{t \times d} \to R^{d \times u}$ vào cuối mô hình nền tảng, trong đó $u = 2$ là số lớp nhãn. Bộ phân loại là một mạng nơ-ron truyền thẳng nhỏ để thực hiện phân loại nhị phân dựa trên trạng thái ẩn $\boldsymbol{h} \in R^{t \times d}$ được tạo bởi mô hình nền tảng:

$$\boldsymbol{h} = \mathcal{F}(\boldsymbol{p}) \tag{1}$$

- Với mô hình BERT, trạng thái ẩn được tính bằng cách lấy trung bình tất cả các token chưa bị che:

$$\boldsymbol{h}_{\text{cls}} = \frac{1}{t} \sum_{i=1}^{t} \boldsymbol{h}_i \tag{2}$$

4. METHOD

4.2 FORMULATION

4.2.2 PROFILE TUNING

- Với các mô hình GPT, chúng tôi sử dụng trạng thái ẩn của token cuối cùng:

$$\boldsymbol{h}_{\text{cls}} = \boldsymbol{h}_t \tag{3}$$

- Cuối cùng, đầu ra được đưa vào bộ phân loại để dự đoán rủi ro tài chính:
- Hàm mất mát được tính bằng Binary Cross Entropy (BCE):
  - $\boldsymbol{y} \in \{0,1\}m$ là nhãn trong training set, $|y| = m$.

$$w_{\text{pos}} = \frac{|\boldsymbol{y}| - \sum_i \boldsymbol{y}_i}{\sum_i \boldsymbol{y}_i} \tag{5}$$

- Loss:

$$\vec{l} = \left\{ -\boldsymbol{y}_i \log(\hat{\boldsymbol{y}}_i) - (1 - \boldsymbol{y}_i) \log(1 - \hat{\boldsymbol{y}}_i) \right\}_{i=1}^{m} \tag{6}$$

- multiply the loss of each positive instance li by the positive weight $w$pos to calculate the weighted BCE loss as follows
-

$$l = \mathcal{L}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = -\frac{1}{m} \sum_{i=1}^{m} [\boldsymbol{y}_i \vec{l}_i w_{\text{pos}} + (1 - \boldsymbol{y}_i) \vec{l}_i] \tag{7}$$

4. METHOD

4.2 FORMULATION

4.2.3 PROFILE TUNING ON MULTIPLE DATASETS

- Given v tabular datasets D_all = {Di}v i=1 = {Xi,$\boldsymbol{y}i$,$\boldsymbol{k}i$}v i=1 and the corresponding profile sets P_all = {Pi}v i=1
- Use all the profiles Pall to tune the large foundation model F and evaluate the performance on each test set of dataset Di.

5. EXPERIMENT

5.1 BASELINES (SO SÁNH)

- Mô hình tăng cường dựa trên cây quyết định: Chúng tôi chọn các mô hình Random Forest , XGBoost, CatBoost và LightGBM làm phương pháp so sánh do hiệu suất vượt trội của chúng trong nhiều bài toán khoa học dữ liệu. Đây là các phương pháp học tập tổ hợp (ensemble learning) hoặc thuật toán tăng cường gradient (gradient boosting) sử dụng cây quyết định.

- **Mạng học sâu**: Lựa chọn bốn mô hình mạng nơ-ron sâu được thiết kế để xử lý dữ liệu bảng như sau:
- **DeepFM** : Một kỹ thuật phổ biến trong ngành công nghiệp, kết hợp máy phân rã ma trận (factorization machines) và học sâu để trích xuất đặc trưng thông qua một kiến trúc mạng nơ-ron mới.
- **STG** : Một framework hỗ trợ đồng thời việc học hàm hồi quy hoặc phân loại phi tuyến và chọn lọc đặc trưng bằng cách sử dụng Stochastic Gates.
- **VIME** : Một áp dụng học tự giám sát và bán giám sát vào dữ liệu bảng bằng cách sử dụng phương pháp ước lượng giá trị thiếu (Value Imputation) và ước lượng mặt nạ (Mask Estimation).
- **TabNet** : Một kiến trúc học sâu có thể diễn giải và đạt hiệu suất cao, tận dụng cơ chế chú ý tuần tự (sequential attention) để chọn ra các đặc trưng quan trọng cho quá trình suy luận trên dữ liệu bảng.

5. EXPERIMENT

5.2 KIẾN TRÚC NỀN TẢNG CỦA FINPT

- Adopt our FinPT to different pretrained foundation models as the backbone model structure.

Model	#Params-All	#Params-T	Trainable Modules
BERT-Base [11]	110M	110M	All params in the model
FinBERT [40]	110M	110M	All params in the model
GPT-2 [29]	117M	117M	All params in the model
T5-Base [30]	220M	220M	All params in the model
Flan-T5-Base [9]	220M	220M	All params in the model
T5-XXL [30]	11B	268M	The last Decoder T5Block
Flan-T5-XXL [9]	11B	260M	The last Decoder T5Block
LLaMA-7B [32]	7B	202M	The last Decoder layer
LLaMA-13B [32]	13B	317M	The last Decoder layer

- Table 2: Experimental settings of backbone foundation mod els for FinPT. “
- #Params-All” : total number of pa rameters in the foundation model. “
- #Params-T” : the number of trainable parameters in the foundation model except for the extra small classifier, which has less than a million parameters.

5. EXPERIMENT
5.3 IMPLEMENT DETAILS

1. Triển khai các mô hình cơ bản (Baseline models)
   - Mô hình cơ bản (baseline models) là các mô hình được sử dụng để so sánh với phương pháp mới đề xuất (ở đây là FinPT). Các mô hình cơ bản này đã được triển khai sẵn và được sử dụng từ các thư viện phổ biến hoặc mã nguồn chính thức.
   - Random Forest: Được triển khai thông qua scikit-learn – một thư viện Python phổ biến cho học máy, giúp xây dựng mô hình Random Forest để phân loại dữ liệu.
   - XGBoost: Sử dụng XGBClassifier từ thư viện XGBoost, một thư viện mạnh mẽ cho học máy, đặc biệt là trong các bài toán phân loại và hồi quy.
   - CatBoost: Dùng CatBoostClassifier từ thư viện CatBoost, một thư viện tối ưu hóa cây quyết định đặc biệt cho dữ liệu phân loại.
   - LightGBM: Áp dụng LGBMClassifier từ thư viện LightGBM, một thư viện mạnh mẽ khác cho bài toán học máy với hiệu suất cao và tốc độ tính toán nhanh.
   - Mạng nơ-ron (Neural network baselines): Các mô hình học sâu như DeepFM, STG, VIME, và TabNet được tái triển khai lại dựa trên mã nguồn mở từ TabSurvey, một công cụ dùng để khảo sát và đánh giá các mô hình học máy cho dữ liệu dạng bảng (tabular data).

5. EXPERIMENT
5.3 IMPLEMENT DETAILS

2. Sử dụng ChatGPT để tạo hồ sơ khách hàng

- Bước 2 trong quy trình chính của nghiên cứu, các hướng dẫn được tạo ra từ dữ liệu bảng (từ Bước 1) sẽ được gửi đến ChatGPT qua API của OpenAI. Quá trình này có thể được mô tả như sau:

```
import openai
response = openai.ChatCompletion.create(
    model="gpt-3.5-turbo",  # Sử dụng mô hình GPT-3.5
    messages=[
        {"role": "system", "content": "You are a helpful financial assistant."},
        {"role": "user", "content": instruction[i]},
    ],  # Danh sách các hướng dẫn từ Bước 1
    temperature=0,  # Điều chỉnh mức độ sáng tạo của phản hồi
)
```

5. EXPERIMENT
5.3 IMPLEMENT DETAILS

3. Tải mô hình ngôn ngữ lớn từ HuggingFace
Bước 3, các mô hình nền tảng lớn (LLMs) như BERT, GPT-2, T5, LLaMA, và Flan-T5 được sử dụng để fine-tune (tinh chỉnh) hồ sơ khách hàng đã được tạo ra từ bước trước. Mô hình này sẽ được tải từ HuggingFace, một nền tảng chứa nhiều mô hình ngôn ngữ lớn.

5. EXPERIMENT

5.4 TRAINING DETAILS

Trong tất cả các thí nghiệm, sử dụng hai GPU NVIDIA A40, mỗi GPU có 48GB bộ nhớ.

Trong tất cả các phiên huấn luyện trên các mô hình so sánh (baseline models), sử dụng kích thước batch là 128 và số epoch tối đa được đặt là 100.
Đối với Profile Tuning trong FinPT, kích thước batch được đặt tối đa là 128 và độ dài chuỗi tối đa là 128, trong đó token đệm (pad) được đặt giống với token kết thúc câu (eos) trong tokenizer. Nếu độ dài chuỗi tối đa (số lượng token) trong một tập dữ liệu lớn hơn 128, thì độ dài chuỗi tối đa của batch sẽ được đặt thành 256, đồng thời kích thước batch sẽ giảm xuống một nửa. Tốc độ học (learning rate) được đặt là 5e−5 và hệ số weight decay là 0,01, sử dụng AdamW [20] làm bộ tối ưu (optimizer).

FinBench cung cấp các tập dữ liệu huấn luyện, xác thực và kiểm tra. Trong quá trình huấn luyện, tất cả các mô hình sử dụng tập huấn luyện để cập nhật tham số và thực hiện đánh giá trên tập xác thực để chọn điểm kiểm tra (checkpoint) tốt nhất. Sau khi huấn luyện, chúng tôi tải điểm kiểm tra tốt nhất và thực hiện kiểm tra trên tập kiểm tra. Mỗi thí nghiệm được thực hiện bốn lần với các hạt ngẫu nhiên khác nhau $\in \{0, 1, 42, 1234\}$, và chúng tôi báo cáo điểm trung bình.

5. EXPERIMENT

5.5 EVALUATION METRICS

Trong tất cả các thí nghiệm trên FinBench, chúng tôi sử dụng F1-score [21] làm tiêu chí đánh giá, vì tất cả các tập dữ liệu trong FinBench đều là bài toán phân loại nhị phân không cân bằng. F1-score phù hợp hơn trong trường hợp này so với độ chính xác (Accuracy), vì độ chính xác có thể dẫn đến tỷ lệ âm tính giả (false negative) cao.

6. KẾT QUẢ VÀ PHÂN TÍCH

6.1 KẾT QUẢ

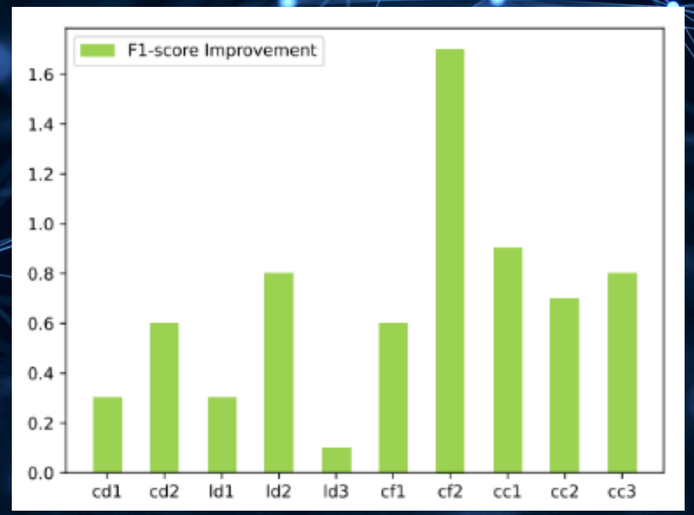
Chúng tôi báo cáo các giá trị F1-score trong dự đoán rủi ro tài chính trên FinBench tại Bảng 3. Trung bình, FinPT vượt trội hơn đáng kể so với các phương pháp dựa trên cây quyết định và các mô hình mạng nơ-ron trước đây, đặc biệt khi chúng tôi áp dụng FinPT để tinh chỉnh hoàn toàn (fully fine-tune) các mô hình nền tảng lớn như GPT-2, T5 và Flan-T5. Dưới đây là phân tích chi tiết về hiệu suất của các mô hình khác nhau trên FinBench.

- Mô hình cây quyết định: CatBoost có hiệu suất trung bình tốt nhất, nhưng các mô hình như Random Forest, XGBoost và LightGBM cũng thể hiện tốt trên một số tập dữ liệu.
- Mô hình mạng nơ-ron: TabNet hoạt động tốt nhất trong tám trên mười tập dữ liệu, nhưng nhìn chung, nhóm mô hình này kém hiệu quả hơn so với phương pháp FinPT và mô hình cây quyết định.
- FinPT (Tinh chỉnh toàn bộ): Flan-T5 là backbone tốt nhất, vượt trội hơn T5 nhờ tinh chỉnh theo hướng dẫn (instruction tuning). GPT-2 hoạt động tốt hơn BERT, cho thấy mô hình Transformer chỉ sử dụng bộ giải mã có thể là bộ phân loại tốt hơn so với bộ mã hóa. FinBERT không khác biệt nhiều so với BERT, cho thấy tiền huấn luyện trên dữ liệu tài chính lớn không mang lại lợi ích rõ ràng.
- FinPT (Tinh chỉnh lớp cuối): LLaMA-13B vượt trội hơn LLaMA-7B và các mô hình nền tảng khác, ngay cả khi chỉ tinh chỉnh lớp giải mã cuối cùng, khẳng định tính hiệu quả của FinPT.

6. KẾT QUẢ VÀ PHÂN TÍCH
6.2. ĐIỀU CHỈNH HỒ SƠ TRÊN TẤT CẢ TẬP DỮ LIỆU

Bảng 3: Kết quả dự đoán rủi ro tài chính trên FinBench. Chúng tôi báo cáo các giá trị F1-score trên cả mười tập dữ liệu trong FinBench và điểm trung bình tổng thể ("Avg"). Trong **Cột "Training"**: **"Grid Search"**: Áp dụng tìm kiếm lưới (grid search) để tối ưu tham số siêu cấp cho các mô hình dựa trên cây quyết định. **"From Scratch"**: Huấn luyện các mô hình mạng nơ-ron cho dữ liệu bảng (NN for Table) từ đầu với khởi tạo trọng số ngẫu nhiên. **"Tune All"**: Tinh chỉnh toàn bộ tham số của các mô hình nền tảng.
"Tune Last": Chỉ tinh chỉnh lớp/cuối cùng của các mô hình nền tảng. Kết quả tốt nhất trên mỗi tập dữ liệu được tô màu **tík,** kết quả tốt nhất của mỗi nhóm mô hình trên từng tập dữ liệu được **gạch chân**.
=> ***Điểm F1 trên Bộ dữ liệu ld3 cải thiện ít nhất chủ yếu vì ld3 chiếm phần lớn điểm chuẩn.***

Model Class	Model	Training	CD1	CD2	LD1	LD2	LD3	CF1	CF2	CC1	CC2	CC3	Avg
Tree-based	RandomForest [16, 18]	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 [8]	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 [27]	Grid Search	21.8	52.2	48.3	72.8	41.4	45.2	21.6	40.5	59.7	65.6	46.91
	LightGBM [17]	Grid Search	21.7	52.6	46.2	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 [36]	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 [41]	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 [2]	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 Tuning 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 [40]	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 [9]	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 Tuning 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 [9]	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 [32]	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 [32]	Tune Last	22.9	52.0	47.2	79.2	42.4	45.7	9.2	41.8	60.4	65.2	46.60



=> SỰ CẢI THIỆN ĐIỂM F1 CỦA FINPT (FLAN-T5-BASE) ĐƯỢC ĐÀO TẠO TRÊN TẤT CẢ CÁC BỘ DỮ LIỆU CÙNG NHAU TRÊN CÁC BỘ DỮ LIỆU RIÊNG BIỆT.

CÁC CHIẾN LƯỢC KHÁC NHAU

Như đã đề cập ở Mục 2, học trong ngữ cảnh (in-context learning - ICL) và tinh chỉnh theo hướng dẫn (instruction tuning - IT) là xu hướng phổ biến trong việc tận dụng các mô hình ngôn ngữ lớn.

Học trong ngữ cảnh (ICL): Cung cấp cho GPT-2, Flan-T5 và LLaMA năm ví dụ theo cấu trúc "Profile: p_i Answer: y_i" ($i \in \{1, 2, 3, 4, 5\}$). Sau đó, sử dụng một prompt để yêu cầu mô hình dự đoán nhãn Yes hoặc No cho một hồ sơ mới (p_t).

Tinh chỉnh theo hướng dẫn (IT): Sử dụng câu lệnh hướng dẫn, chẳng hạn: "Dự đoán xem khách hàng sau đây có rủi ro tài chính hay không. p_t Option 1: Yes. Option 2: No. Answer:"

Mô hình nền tảng thường gặp vấn đề ảo giác (hallucination) và né tránh đưa ra dự đoán chắc chắn. Do đó, đầu ra của ICL và IT chỉ được coi là đúng nếu chứa nhãn "Yes" hoặc "No" trong văn bản. Dù tiêu chí đánh giá này khá thoáng, cả hai phương pháp vẫn cho hiệu suất kém (< 10 F1-score) so với các phương pháp phân loại khác. Tuy nhiên, đầu ra từ ICL và IT vẫn có giá trị trong việc cung cấp giải thích giúp con người phân tích kết quả. Ví dụ:

"Dựa trên thông tin hiện có, khách hàng này có vẻ không có rủi ro tài chính. Cô ấy có số dư cao, thu nhập tốt và đã là khách hàng trong một khoảng thời gian dài với trạng thái giao dịch là 1.0. Ngoài ra, loại tín dụng của cô ấy ở mức trung bình – không xuất sắc nhưng cũng không tệ. Vì vậy, Option 2: No, cô ấy không có rủi ro tài chính, có vẻ là câu trả lời hợp lý hơn."

THANK YOU