深入解析論文: "Revisiting Deep Learning Models for Tabular Data"

Introduction

這部分討論了深度學習在表格數據上應用的現狀與問題。

主要問題陳述

儘管已有大量針對表格數據的深度學習模型被提出，但這些模型通常沒有被適當地相互比較，且現有的研究往往使用不同的基準和實驗協議。因此，對於研究人員和實踐者來說，哪些模型表現最佳仍然不明確。此外，該領域仍然缺乏有效的基準模型——即能夠在不同問題上提供競爭性能的易用模型。

研究目標

論文作者希望通過以下方式提高表格數據深度學習的基準水平：

1. 對表格數據深度學習架構的主要家族進行概述
2. 識別兩種簡單而強大的深度架構作為基準

提出的解決方案

作者提出了兩個模型：

1. 類似 ResNet 的架構，這成為一個強大的基準，在先前的工作中經常被忽視
2. FT-Transformer：作者對 Transformer 架構為表格數據所做的簡單改編，在大多數任務上優於其他解決方案

研究貢獻

論文做出了四個主要貢獻：

1. 在多樣化的任務上徹底評估表格深度學習主要模型，以研究它們的相對性能
2. 證明簡單的 ResNet 類架構是表格深度學習的有效基準，但被現有文獻忽視。由於其簡單性，作者建議在未來的表格深度學習工作中將其用作比較基準
3. 引入 FT-Transformer，一種用於表格數據的 Transformer 架構簡單適配，成為該領域的新強大解決方案。作者觀察到它是一種更通用的架構：它

在比其他深度學習模型更廣泛的任務上表現良好

4. 揭示 GBDT 和深度模型之間仍然沒有普遍優越的解決方案

Related Work

這部分回顧了表格數據處理的現有技術，特別關注梯度提升決策樹(GBDT)和不同類型的深度學習模型。

梯度提升決策樹(GBDT)

對於表格數據問題，當前"淺層"最先進技術是決策樹集成，如 GBDT(梯度提升決策樹)，這通常是各種機器學習競賽中的首選。目前有幾個已建立的 GBDT 庫，如 XGBoost、LightGBM、CatBoost，這些被研究人員和實踐者廣泛使用。雖然這些實現在細節上有所不同，但在大多數任務上，它們的性能差異不大。

深度學習模型分類

論文將深度學習模型分為三大類：

1. 可微分決策樹

這類模型受決策樹集成在表格數據上的強大性能啟發。由於決策樹不可微分且不允許梯度優化，它們不能用作以端到端方式訓練的管道的組件。為解決這個問題，一些工作提出"平滑"內部樹節點中的決策函數，使整體樹函數和樹路由可微分。雖然這一系列的方法可以在某些任務上優於 GBDT，但在作者的實驗中，它們並不能始終超越 ResNet。

這類模型的主要思想是讓決策樹變得可微分，從而能夠用於端到端的深度學習系統中。例如，NODE(Neural Oblivious Decision Ensembles)將決策樹中的硬決策邊界轉換為軟的、可微分的決策函數。

2. 基於注意力的模型

由於基於注意力的架構在不同領域的普遍成功，一些作者也提出將類似注意力的模塊用於表格深度學習。在作者的實驗中，他們表明適當調優的 ResNet 優於現有的基於注意力的模型。

不過，作者也發現了將 Transformer 架構應用於表格數據的有效方法：由此產生的架構在大多數任務上優於 ResNet。

3. 顯式建模乘法交互

在推薦系統和點擊率預測的文獻中，一些研究批評 MLP 不適合建模特徵之間的乘法交互。受此動機啟發，一些工作提出了將特徵乘積納入 MLP 的不同方法。然而，在作者的實驗中，這些方法並不優於適當調優的基準模型。

這類模型強調特徵之間的交互和相乘關係，如 DCN(Deep & Cross Network)，它們嘗試通過特徵交叉來捕捉更複雜的模式。

其他架構

文獻中還提出了一些無法明確分配到上述任何一組的其他架構設計。總體而言，研究界已經開發了各種模型，它們在不同的基準上進行評估，並且很少相互比較。

Models for Tabular Data Problems

這部分介紹了論文中主要關注的深度學習架構，包括現有解決方案和作者提出的新模型。

符號定義

論文考慮監督學習問題，$D=\{(x_i, y_i)\}^n_{i=1}$表示數據集，其中 $x_i=(x^{(num)}_i, x^{(cat)}_i)\in X$ 表示物件的數值特徵 $x^{(num)}_{ij}$ 和類別特徵 $x^{(cat)}_{ij}$，$y_i\in Y$ 表示相應的物件標籤。特徵總數表示為 k。數據集分為三個不相交子集：$D = D_{train} \cup D_{val} \cup D_{test}$，其中 D_{train} 用於訓練，D_{val} 用於早停和超參數調整，D_{test} 用於最終評估。論文考慮三種任務類型：二元分類 $Y = \{0, 1\}$，多類分類 $Y = \{1, ..., C\}$和回歸 $Y = R$。

MLP 模型

MLP 架構在方程 1 中被形式化：

MLP(x) = Linear (MLPBlock (... (MLPBlock(x)))) MLPBlock(x) = Dropout(ReLU(Linear(x)))

這是一個標準的多層感知器，由線性層、ReLU 激活函數和 Dropout 層組成。

ResNet 模型

作者了解到有一次嘗試設計類似 ResNet 的基準（Klambauer 等，2017），但報告的結果不具競爭力。然而，鑑於 ResNet 在計算機視覺中的成功故事和它最近在 NLP 任務上的成就，作者決定再次嘗試，並按照方程 2 中的描述構建了一個簡單的 ResNet 變體。主要構建模塊相比原始架構進行了簡化，從輸入到輸

出有一條幾乎清晰的路徑，作者發現這對優化有益。總體而言，他們期望這種架構在需要更深層表示的任務上優於 MLP。

ResNet(x) = Prediction (ResNetBlock (... (ResNetBlock (Linear(x)))))
ResNetBlock(x) = x + Dropout(Linear(Dropout(ReLU(Linear(BatchNorm(x))))))
Prediction(x) = Linear (ReLU (BatchNorm (x)))

ResNet 的核心思想是使用跳躍連接，使深層網絡更容易訓練。對於表格數據，作者提出的 ResNet 變體保留了這種跳躍連接的思想，但進行了一些調整以更好地適應表格數據的特性。

FT-Transformer 模型

這部分介紹了 FT-Transformer（Feature Tokenizer + Transformer）—— 一種用於表格域的 Transformer 架構的簡單適配。圖 1 展示了 FT-Transformer 的主要部分。

FT-Transformer 由兩個主要部分組成：

1. Feature Tokenizer：將所有特徵（類別和數值）轉換為嵌入
2. Transformer：應用一堆 Transformer 層於嵌入上

簡而言之，該模型將所有特徵（類別和數值）轉換為嵌入，並對嵌入應用一堆 Transformer 層。因此，每個 Transformer 層在一個物件的特徵級別上操作。

Feature Tokenizer

Feature Tokenizer 模塊將輸入特徵 x 轉換為嵌入 $T \in R^{kxd}$。給定特徵 xj 的嵌入計算如下：

$T_j = b_j + f_j(x_j) \in R^d \quad f_j: X_j \to R^d$

其中 bj 是第 j 個特徵偏置，對於數值特徵，$f^{(num)}_j$ 實現為與向量 $W^{(num)}_j \in R^d$ 的元素級乘法，對於類別特徵，$f^{(cat)}_j$ 實現為查找表 $W^{(cat)}_j \in R^{s_j xd}$。總體而言：

$T^{(num)}_j = b^{(num)}_j + x^{(num)}_j \cdot W^{(num)}_j \in R^d \quad T^{(cat)}_j = b^{(cat)}_j + e^T_j W^{(cat)}_j \in R^d \quad T = \text{stack}[T^{(num)}_1, ..., T^{(num)}_{k(num)}, T^{(cat)}_1, ..., T^{(cat)}_{k(kat)}] \in R^{kxd}$

其中 e^T_j 是相應類別特徵的 one-hot 向量。

Transformer

[CLS]標記（或"分類標記"，或"輸出標記"）的嵌入被添加到 T 中，並應用 L 個 Transformer 層 $F_1, ..., F_l$：

$T_0 = \text{stack}[[CLS], T]$ $T_i = F_i(T_{i-1})$

在第 5 頁中進一步說明： 作者使用 PreNorm 變體以便更易於優化，見圖 2。在 PreNorm 設置中，作者還發現有必要從第一個 Transformer 層中移除第一個規範化以達到良好的性能。

預測層

[CLS]標記的最終表示用於預測: ŷ = Linear(ReLU(LayerNorm(T^[CLS]_L)))

限制

FT-Transformer 相比簡單模型（如 ResNet）需要更多資源（硬件和時間）進行訓練，當特徵數量"太大"時（由可用硬件和時間預算決定）可能不易擴展。因此，FT-Transformer 的廣泛使用可能導致 ML 管道產生更多 CO_2排放，因為表格數據問題無處不在。描述問題的主要原因在於 vanilla MHSA（多頭自注意力）關於特徵數量的二次複雜度。然而，這個問題可以通過使用 MHSA 的高效近似來緩解。此外，仍然可以將 FT-Transformer 蒸餾到更簡單的架構中以獲得更好的推理性能。

其他模型

論文還包括了幾個專為表格數據設計的現有模型進行比較：

- SNN (Klambauer 等, 2017)：具有 SELU 激活的類 MLP 架構，使更深層模型能夠訓練。
- NODE (Popov 等, 2020)：可微分的顯式決策樹集成。
- TabNet (Arik 和 Pfister, 2020)：一種循環架構，交替進行特徵的動態重新加權和常規前饋模塊。
- GrowNet (Badirli 等, 2020)：梯度提升弱 MLP。官方實現僅支持分類和回歸問題。
- DCN V2 (Wang 等, 2020a)：包含一個類 MLP 模塊和特徵交叉模塊（線性層和乘法的組合）。
- AutoInt (Song 等, 2019)：將特徵轉換為嵌入並對嵌入應用一系列基於注

意力的轉換。

- XGBoost (Chen 和 Guestrin, 2016)：最流行的 GBDT 實現之一。
- CatBoost (Prokhorenkova 等, 2018)：使用顯式決策樹作為弱學習器的 GBDT 實現。

Experiments

這部分詳細說明了論文中的實驗設計、數據集選擇以及結果分析。

比較範圍

作者解釋他們專注於不同架構的相對性能，而不採用各種模型無關的深度學習實踐，如預訓練、額外損失函數、數據增強、蒸餾、學習率預熱、學習率衰減等。雖然這些實踐可能提高性能，但作者的目標是評估不同模型架構施加的歸納偏差的影響。

數據集

作者使用了十一個公共數據集（詳細描述見補充材料）。對於每個數據集，有一個確切的訓練-驗證-測試分割，所以所有算法使用相同的分割。

數據集包括： California Housing（CA，房地產數據），Adult（AD，收入估計），Helena（HE，匿名數據集），Jannis（JA，匿名數據集），Higgs（HI，模擬物理粒子），ALOI（AL，圖像），Epsilon（EP，模擬物理實驗），Year（YE，音頻特徵），Covertype（CO，森林特徵），Yahoo（YA，搜索查詢），Microsoft（MI，搜索查詢）。作者按照逐點方法對排序問題（Microsoft，Yahoo）進行學習，將其視為回歸問題。

實現細節

數據預處理

數據預處理對深度學習模型至關重要。對於每個數據集，所有深度模型使用相同的預處理以進行公平比較。默認情況下，作者使用 Scikit-learn 庫中的分位數轉換。他們對 Helena 和 ALOI 進行標準化（減去均值和縮放）。後者表示圖像數據，標準化是計算機視覺中的常見做法。在 Epsilon 數據集上，作者觀察到預處理對深度模型的性能有害，所以在這個數據集上使用原始特徵。作者對所有算法的回歸目標應用標準化。

調參

對於每個數據集，作者仔細調整每個模型的超參數。最佳超參數是在驗證集上表現最好的那些，因此測試集從不用於調整。對於大多數算法，作者使用 Optuna 庫進行貝葉斯優化（Tree-Structured Parzen Estimator 算法），據報導優於隨機搜索。對於其他算法，作者遍歷相應論文推薦的預定義配置集。

評估

對於每個調優的配置，作者運行 15 個不同隨機種子的實驗，並報告測試集上的性能。對於一些算法，作者還報告了沒有超參數調整的默認配置的性能。

集成

對於每個模型，在每個數據集上，作者通過將 15 個單一模型分成三個大小相等的不相交組，並平均每組內單一模型的預測來獲得三個集成。

神經網絡

作者對分類問題最小化交叉熵，對回歸問題最小化均方誤差。對於 TabNet 和 GrowNet，作者遵循原始實現並使用 Adam 優化器。對於所有其他算法，使用 AdamW 優化器。作者不應用學習率調度。對於每個數據集，除非在相應論文中給出批量大小的特殊指示，否則所有算法使用預定義的批量大小。作者持續訓練，直到在驗證集上有 patience + 1 個連續周期沒有改進；對所有算法設置 patience = 16。

類別特徵

對於 XGBoost，作者使用 one-hot 編碼。對於 CatBoost，採用內置的類別特徵支持。對於神經網絡，對所有類別特徵使用相同維度的嵌入。

深度學習模型比較結果

表 2 報告了深度架構的結果。

主要發現：

- MLP 仍然是一個良好的理智檢查
- ResNet 成為一個有效的基準，沒有一個競爭者能始終超越它
- FT-Transformer 在大多數任務上表現最佳，成為該領域的新強大解決方

案

- 調優使簡單模型如 MLP 和 ResNet 具有競爭力，因此作者建議可能時調優基準。幸運的是，通過諸如 Optuna 之類的庫，今天它更容易接近。

在其他模型中，NODE 是唯一一個在多個任務上表現出高性能的模型。然而，它在六個數據集（Helena、Jannis、Higgs、ALOI、Epsilon、Covertype）上仍然劣於 ResNet，同時是更複雜的解決方案。此外，它不是真正的"單一"模型；事實上，它通常包含明顯多於 ResNet 和 FT-Transformer 的參數，並具有類似集成的結構。

深度學習模型和 GBDT 的比較

這部分的目標是檢查深度學習模型在概念上是否準備好超越 GBDT。為此，作者比較了使用 GBDT 或深度學習模型可以達到的最佳指標值，不考慮速度和硬件要求（無疑，GBDT 是更輕量級的解決方案）。

作者通過比較集成而非單一模型來實現這一點，因為 GBDT 本質上是一種集成技術，作者預期深度架構將從集成中受益更多。

默認超參數結果

作者首先使用默認配置來檢查"開箱即用"的性能，這是一個重要的實際場景。默認 FT-Transformer 意味著所有超參數設置為作者在補充材料中提供的特定值。表 4 顯示 FT-Transformers 的集成大多優於 GBDT 的集成，只有兩個數據集（California Housing、Adult）除外。有趣的是，默認 FT-Transformers 的集成與調優的 FT-Transformers 的集成表現相當。

主要發現：FT-Transformer 允許開箱即用地構建強大的集成。

調優超參數結果

一旦超參數得到適當調整，GBDT 開始在一些數據集（California Housing、Adult、Yahoo）上占主導地位。在這些情況下，差距足夠顯著，可以得出結論，深度學習模型並不普遍優於 GBDT。重要的是，深度學習模型在大多數任務上優於 GBDT 的事實並不意味著深度學習解決方案在任何意義上"更好"。事實上，這只意味著構建的基準對"深度學習友好"問題略有偏向。

誠然，GBDT 仍然不適合具有大量類別的多類問題。根據類別數量，GBDT 可能表現不盡如人意（Helena）或由於訓練極其緩慢而無法調優（ALOI）。

主要發現：

- 在深度學習模型和 GBDT 中仍然沒有通用解決方案
- 旨在超越 GBDT 的深度學習研究努力應該集中在 GBDT 優於最先進深度學習解決方案的數據集上。請注意，包含"深度學習友好"問題仍然很重要，以避免在此類問題上性能下降。

FT-Transformer 的有趣特性

表 4 講述了另一個重要故事。即，FT-Transformer 相對於以 ResNet 形式的"傳統"深度學習模型的大部分優勢，正是在 GBDT 優於 ResNet 的那些問題上（California Housing、Adult、Covertype、Yahoo、Microsoft），而在其餘問題上與 ResNet 表現相當。換句話說，FT-Transformer 在所有任務上都提供了競爭性能，而 GBDT 和 ResNet 只在任務的某些子集上表現良好。這一觀察可能是 FT-Transformer 是表格數據問題的更"通用"模型的證據。

Analysis

這部分深入分析了 FT-Transformer 與 ResNet 性能差異的原因，並探討了 FT-Transformer 架構的特定設計選擇的影響。

FT-Transformer 何時優於 ResNet?

在這部分，作者邁出了理解 FT-Transformer 和 ResNet 之間行為差異的第一步，這種差異首次在 4.6 節中被觀察到。為了實現這一目標，作者設計了一系列合成任務，兩個模型的性能差異從微不足道逐漸變為巨大。

作者生成並固定物件$\{x_i\}_{i=1}^{n}$，執行一次訓練-驗證-測試分割，並在兩個回歸目標之間進行插值：fGBDT，假設對 GBDT 更容易，和 fDL，預期對 ResNet 更容易。正式來說，對於一個物件：

$x \sim N(0, I_k)$, $y = \alpha \cdot f_{GBDT}(x) + (1 - \alpha) \cdot f_{DL}(x)$

其中 fGBDT(x)是 30 個隨機構建的決策樹的平均預測，fDL(x)是具有三個隨機初始化隱藏層的 MLP。

圖 3 中的結果顯示，ResNet 和 FT-Transformer 在 ResNet 友好任務上表現同樣出色，並在這些任務上優於 CatBoost。然而，當目標變得更 GBDT 友好時，ResNet 的相對性能顯著下降。相比之下，FT-Transformer 在整個任務範圍內都

提供競爭性能。

進行的實驗揭示了一類被 FT-Transformer 比 ResNet 更好地近似的函數。此外，這些函數基於決策樹的事實與 4.6 節中的觀察和表 4 中的結果相關，在表 4 中，FT-Transformer 正是在 GBDT 優於 ResNet 的那些數據集上顯示了對 ResNet 最令人信服的改進。

這表明 FT-Transformer 有能力捕捉樹型決策結構，這是 GBDT 的核心優勢，同時保持深度學習模型的優點。

消融研究

這部分測試了 FT-Transformer 的一些設計選擇。

首先，作者將 FT-Transformer 與 AutoInt 比較，因為它在精神上是最接近的競爭者。AutoInt 也將所有特徵轉換為嵌入並在其上應用自注意力。

然而，在細節上，AutoInt 與 FT-Transformer 顯著不同：其嵌入層不包括特徵偏置，其骨幹與 vanilla Transformer 顯著不同，且推理機制不使用[CLS]標記。

其次，作者檢查 Feature Tokenizer 中的特徵偏置是否對良好性能至關重要。

Analysis (繼續)

消融研究 (續)

作者按照與 4.3 節相同的協議調整和評估不帶特徵偏置的 FT-Transformer，並重用表 2 中的剩餘數字。表 5 中展示了在 15 次運行中平均的結果，這證明了 Transformer 骨幹相對於 AutoInt 的優越性和特徵偏置的必要性。

實驗結果顯示，完整的 FT-Transformer 在所有測試數據集上都優於 AutoInt 和不帶特徵偏置的 FT-Transformer。例如，在 California Housing 數據集上，FT-Transformer 的 RMSE 為 0.459，而沒有特徵偏置的版本為 0.470，AutoInt 為 0.474。這表明 Feature Tokenizer 中的特徵偏置是模型良好性能的關鍵組成部分。

從注意力圖獲取特徵重要性

作者評估了注意力圖作為給定樣本集的 FT-Transformer 特徵重要性的信息來源。對於第 i 個樣本，他們計算 Transformer 前向傳遞中[CLS]標記的平均注意力圖 pi。然後，將獲得的個體分佈平均為一個分佈 p，表示特徵重要性:

p = (1/nsamples) Σi pi pi = (1/(nheads × L)) Σh,l pihl

其中 pihl 是從第 i 個樣本上第 l 層的前向傳遞中第 h 個頭的[CLS]標記的注意力圖。所描述的啟發式技術的主要優勢是其效率：它只需要一個樣本的單一前向傳遞。

為了評估這種方法，作者將其與積分梯度（IG，Sundararajan 等，2017）進行比較，後者是一種適用於任何可微分模型的通用技術。作者使用排列測試（PT，Breiman，2001）作為合理的可解釋方法，允許建立秩相關的構建性度量。他們在訓練集上運行所有方法，並在表 6 中總結結果。

有趣的是，所提出的方法產生了合理的特徵重要性，並表現與 IG 相似（注意這並不意味著與 IG 的特徵重要性相似）。考慮到 IG 可能慢幾個數量級，而 PT 形式的"基線"需要(nfeatures + 1)次前向傳遞（相比於所提出方法只需一次），作者得出結論，簡單平均注意力圖在成本效益方面可能是一個不錯的選擇。

這表明 FT-Transformer 不僅在預測性能上有優勢，還可以通過分析其注意力機制提供模型的可解釋性，這對實際應用非常有價值。利用注意力圖來獲取特徵重要性可以幫助理解模型決策的依據，而且計算成本遠低於其他方法。

Conclusion

本文調查了表格數據深度學習領域的現狀，並改進了表格深度學習的基準狀態。作者通過實驗展示了幾個關鍵發現：

1. 簡單的 ResNet 類架構可以作為一個有效的基準 這個發現挑戰了之前的觀念，即只有複雜的特定領域架構才能處理好表格數據。作者證明，通過適當調整，一個簡單的 ResNet 變體可以提供強大的性能，並建議將它作為未來表格深度學習工作的比較基準。

2. FT-Transformer 是一個簡單但強大的新解決方案 作者提出的 FT-Transformer 模型在大多數任務上優於其他深度學習解決方案。這個模型是對 Transformer 架構的簡單適配，通過 Feature Tokenizer 將表格數據轉換為適合 Transformer 處理的格式。實驗表明，FT-Transformer 在各種表格數據任務上表現出色。

3. FT-Transformer 是一個更通用的解決方案 特別值得注意的是，FT-Transformer 表現出更高的通用性 - 它在更廣泛的任務類型上表現良好，包括那些 GBDT 通常優於傳統深度學習模型的任務。這表明它結合

了深度學習和決策樹模型的優勢。

4. GBDT 和深度模型之間仍然沒有絕對優勝者 作者將最佳深度學習模型與 GBDT 進行比較，發現沒有一種方法可以在所有任務上始終優於另一種。這強調了繼續研究這兩類方法的重要性，並根據具體問題選擇適當的模型。

這項研究為表格數據的深度學習提供了新的基準和見解。作者開源了所有代碼和研究細節，希望他們的評估和兩個簡單模型（ResNet 和 FT-Transformer）將作為表格深度學習進一步發展的基礎。

總結性例子：

- 對於需要處理許多數值和類別特徵的金融預測問題，FT-Transformer 能夠同時捕捉複雜的特徵交互和層次結構，提供更準確的預測
- 在醫療診斷應用中，ResNet 提供了一個簡單但強大的基準，能夠處理混合類型的患者數據
- 對於客戶流失預測等任務，GBDT 可能在某些數據集上仍然表現最佳，但 FT-Transformer 可能更好地適應不同形式的數據分布

這項研究的主要貢獻是提供了一個統一的比較框架，引入了新的強大基準模型，並為選擇適當的模型處理不同類型的表格數據問題提供了指導。