Applications of Deep Learning to Tabular Datasets

11320 IEEM 513600

Deep Learning for Industrial Applications

2025/03/13 Ming Chung Lim

國立清華大學
NATIONAL TSING HUA UNIVERSITY

Tabular Dataset

Background Survey for Deep Learning — Edited

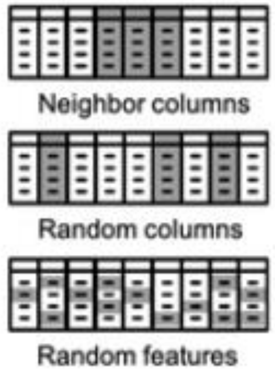
Sheet 1

時間戳記	4. Department and year	5. Your undergraduate major	6. A short Introduction to your research field or research interest
2024/02/22 3:37:01 下午 GMT+8	IEEM master	Statistics	Deep learning
2024/02/22 3:38:38 下午 GMT+8	工工碩一	工工	最佳化
2024/02/22 3:40:57 下午 GMT+8	工工碩一	清大工工	目前在研究分析脈波預測糖尿病
2024/02/22 3:41:04 下午 GMT+8	IEEM 2025	IEEM	AI applied to production line improvement
2024/02/22 3:41:30 下午 GMT+8	工工碩一	運輸與物流管理學系	整數賽局
2024/02/22 3:42:12 下午 GMT+8	工工系碩一	工工系	數學模型建模
2024/02/22 3:42:38 下午 GMT+8	工工所碩一	運輸科學系	賽局理論
2024/02/22 3:44:33 下午 GMT+8	工業工程與工業管理學系 碩一	企業管理學系	碳權相關的預測
2024/02/22 3:44:40 下午 GMT+8	工工碩一	工工	Machine Learning, Computer Vision
2024/02/22 3:47:30 下午 GMT+8	112工工	工工	Machine learning
2024/02/22 3:50:06 下午 GMT+8	交大工管所 碩一	交大工工系	Abnomaly detection, RUL prediction, imbalanced learning
2024/02/22 4:38:11 下午 GMT+8	工工碩一	工工	用演算法協助廠商搜索潛在上下游客戶
2024/02/22 4:44:49 下午 GMT+8	工工系碩一	工工系	Operations research, machine learning
2024/02/22 5:13:35 下午 GMT+8	工工碩一	IEEM	Machine learning 、computer vision
2024/02/22 5:21:05 下午 GMT+8	交大工業工程與管理學系 碩士班 一年級	交大 運輸與物流管理學系 輔系 工工系	學習如何利用深度強化學習的相關知識，應用於工業上的排程問題
2024/02/22 5:30:42 下午 GMT+8	iPHD 博二	Engineering Science	Digital Transformation and Sustainability in Taiwan Small & Medium-sized Enterprises
2024/02/22 6:23:27 下午 GMT+8	工工所博士班一年級	工業工程與工程管理	Incremental learning
2024/02/22 6:32:27 下午 GMT+8	工工所人因組碩一	工業設計	目前在實驗室的研究領域是跟調度分配有關。 前一陣子讀了一篇叫車出行的訂單調度分配研究，裡面提到他們是使用深度學習在實
2024/02/22 8:38:05 下午 GMT+8	iphd year 1	Mechanical Engineering	AI ^ Entrepreneurship
2024/02/22 9:38:07 下午 GMT+8	工工所碩一	清大工工系	目前還沒有研究特定領域，但之後可能會往機器視覺相關或是混合演算法方面進行研究

Features

The tabular data commonly used in many fields such as healthcare, advertisement, finance, and law.

- **Structured Format**: Tabular data is organized into rows and columns, with rows representing records and columns representing variables.
- **Heterogeneous Data Types**: Columns can contain different data types, including numerical, categorical, datetime, and text.
- **Feature Relationships**: Features within tabular datasets may exhibit complex relationships and dependencies that are crucial for model accuracy.
- **Missing Values**: Tabular datasets often contain missing values, necessitating strategies like imputation or omission for effective data analysis.



 Source: https://openreview.net/pdf?id=vrhNQ7aYSdr

Use Pandas to Process

lab - JupyterLab

conda: jlab_server

File Edit View Run Kernel Tabs Settings Help

Untitled2.ipynb

Code

Python 3 (ipykernel)

[1]:
```python
import pandas as pd
```

[2]:
```python
df = pd.read_csv('/Users/taco/Downloads/archive-2/sp500_stocks.csv')
```

[3]:
```python
df
```

[3]:

	Date	Symbol	Adj Close	Close	High	Low	Open	Volume
0	2010-01-04	MMM	59.318886	83.019997	83.449997	82.669998	83.089996	3043700.0
1	2010-01-05	MMM	58.947342	82.500000	83.230003	81.699997	82.800003	2847000.0
2	2010-01-06	MMM	59.783295	83.669998	84.599998	83.510002	83.879997	5268500.0
3	2010-01-07	MMM	59.826176	83.730003	83.760002	82.120003	83.320000	4470100.0
4	2010-01-08	MMM	60.247749	84.320000	84.320000	83.300003	83.690002	3405800.0
...	...	...	...	...	...	...	...	...
1794196	2024-02-29	ZTS	198.330002	198.330002	200.529999	198.190002	200.500000	2849200.0
1794197	2024-03-01	ZTS	190.820007	190.820007	196.229996	188.490005	196.000000	3724300.0
1794198	2024-03-04	ZTS	187.860001	187.860001	190.419998	186.509995	189.610001	3757200.0
1794199	2024-03-05	ZTS	185.229996	185.229996	187.729996	184.419998	187.440002	2508500.0
1794200	2024-03-06	ZTS	183.529999	183.529999	187.240005	183.229996	185.300003	5028502.0

1794201 rows × 8 columns

[4]:
```python
df.isnull().sum()
```

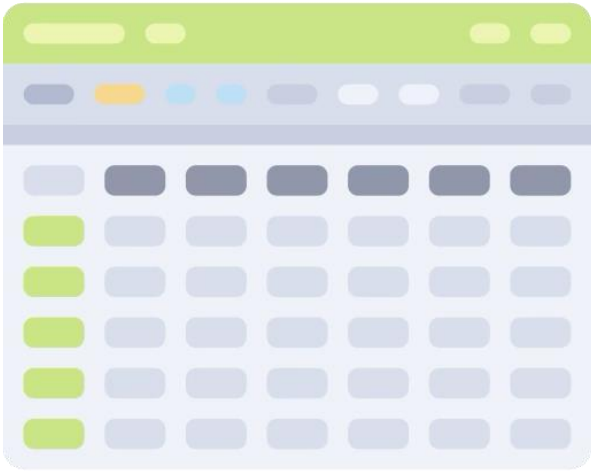
[4]:
```
Date              0
```

Use Pandas to Process

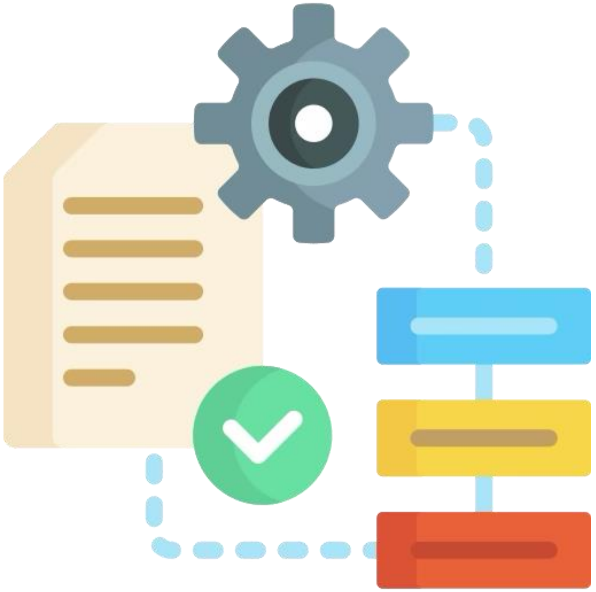
	match_datetime	league_id	league_name	home_id	home_name	away_id	away_name	stadium_id	stadium_name	season	...	ht_ou_betco
11693	NaN	NaN	NaN	2019	1052407	2	27497030	NaN	b'\x01'	2018-11-26 22:23:04	...	
11694	2018-11-26 20:30:00	64.0	Premier League	1295	Burnley FC	513	Newcastle United FC	NaN	NaN	NaN	...	
11695	NaN	NaN	NaN	2019	1052407	2	27497030	NaN	b'\x01'	2018-11-26 22:23:05	...	
11696	2018-11-26 20:30:00	64.0	Premier League	1295	Burnley FC	513	Newcastle United FC	NaN	NaN	NaN	...	
11697	NaN	NaN	NaN	2019	1052407	2	27497030	NaN	b'\x01'	2018-11-26 22:23:06	...	

Must be careful about the NaN values!

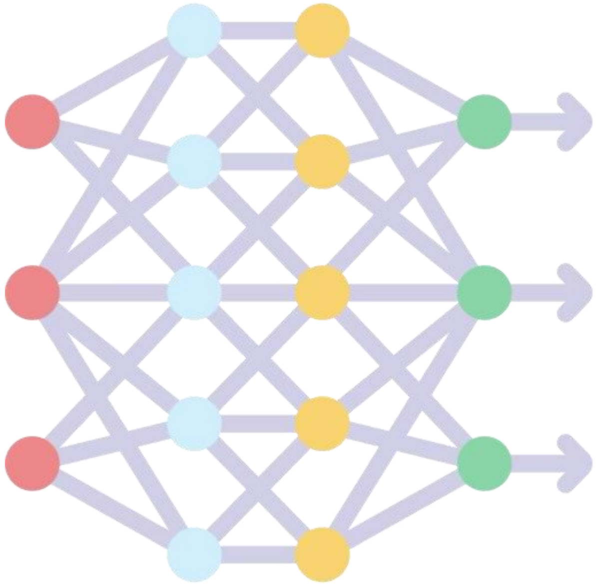
3 Steps for Model Development



Know Your Data



Process Data



Build a Model

Why Deep Learning for Tabular Data?

Traditional ML models (XGBoost, LightGBM) dominate tabular data tasks, but deep learning offers unique advantages:

- **Capturing Complex Nonlinear Relationships:**
  - Deep learning models, particularly neural networks, can approximate highly nonlinear functions without requiring explicit feature engineering.
  - In high-dimensional data, they can automatically discover intricate interactions between features.
- **Automated Feature Learning:**
  - Traditional models need manual feature engineering.
  - Deep learning learns feature representations from raw data.
  - Embeddings (e.g., TabTransformer) help with categorical features.

Source: https://ieeexplore.ieee.org/abstract/document/9998482

Why Deep Learning for Tabular Data?

Traditional ML models (XGBoost, LightGBM) dominate tabular data tasks, but deep learning offers unique advantages:

- **Data Generation for Imbalanced & Privacy-Constrained Data:**
  - GANs (CTGAN, PATE-GAN) and VAEs (TVAE) can generate synthetic tabular data to address class imbalance by generating more samples for underrepresented categories.
  - Enable privacy-preserving machine learning by synthesizing realistic data without exposing sensitive information.
- **Multimodal Learning:**
  - Many real-world applications involve a mix of structured tabular data + unstructured data (e.g., text, images, time series).
  - Deep learning can naturally integrate multimodal inputs, making it a powerful tool for finance, healthcare, and recommendation systems.

Source: https://ieeexplore.ieee.org/abstract/document/9998482

Challenges of Deep Learning for Tabular Data

Deep learning method usually not as effective as GBDT for tabular data. (Why?)

- **Heterogeneous Data Types:** Numerical, categorical, datetime, text → Requires extensive preprocessing.
- **Weak Feature Relationships**: Unlike images or text, tabular data lacks spatial or sequential dependencies.
- **High Preprocessing Dependency**: Categorical features need encoding (One-Hot, Target Encoding, Embedding). Missing values must be handled properly.
- **Limited Performance vs. GBDT**: XGBoost, LightGBM still dominate small-to-medium-sized tabular datasets. DNNs require much more data to generalize effectively.
- **Interpretability Issues**: Many industries (finance, healthcare) require explainable AI, but deep learning models are often "black boxes."

Source: https://ieeexplore.ieee.org/abstract/document/9998482

Deep Learning Methods for Tabular Data

How do deep learning methods try to overcome these challenges?

- **Data Transformation Approaches:**
  - Encoding categorical data (One-Hot, Target Encoding, Embeddings).
  - Converting tabular data into images (SuperTML, IGTD).
- **Specialized Architectures:**
  - Hybrid Models: NODE (Neural Oblivious Decision Trees), DeepGBM (GBDT + DNN).
  - Transformer-based Models: TabNet (Uses sparse attention for feature selection), TabTransformer (SAINT: Self-attention for feature relationships).
- **Regularization Strategies:**
  - RLN (Row-wise Learning Normalization) to prevent overfitting.
  - Batch Normalization, Dropout for training stability.

 Source: https://ieeexplore.ieee.org/abstract/document/9998482

Benchmark – GBDT vs. Deep Learning

- **Empirical Comparisons from Research:**
  - GBDT (XGBoost, LightGBM, CatBoost) still outperform deep learning on most small-to-medium-sized tabular datasets.
  - Transformer-based models (SAINT, TabTransformer) show promise on large-scale datasets but require more computational resources.
  - Training Cost Comparison:
    - GBDT → Fast training, works well with limited data.
    - Deep Learning → Requires large datasets, GPU acceleration, and extensive tuning.
- **When to Use Which?**
  - If dataset is small, structured, and requires interpretability → Use GBDT.
  - If dataset is large, complex, and feature relationships are hard to capture → Consider DNN-based models.

Source: https://ieeexplore.ieee.org/abstract/document/9998482

Benchmark – GBDT vs. Deep Learning

OPEN PERFORMANCE BENCHMARK RESULTS BASED ON (STRATIFIED) FIVEFOLD CROSS VALIDATION. WE USE THE SAME FOLD SPLITTING STRATEGY FOR EVERY DATASET. THE TOP RESULTS FOR EACH DATASET ARE IN **BOLD**, WE ALSO UNDERLINE THE SECOND-BEST RESULTS. THE MEAN AND STANDARD DEVIATION VALUES ARE REPORTED FOR EACH BASELINE MODEL. MISSING RESULTS INDICATE THAT THE CORRESPONDING MODEL COULD NOT BE APPLIED TO THE TASK TYPE (REGRESSION OR MULTICLASS CLASSIFICATION)

	Method	HELOC		Adult		HIGGS		Covertype		Cal. Housing
		Acc ↑	AUC ↑	Acc ↑	AUC ↑	Acc ↑	AUC ↑	Acc ↑	AUC ↑	MSE ↓
Machine Learning	Linear Model	73.0±0.0	80.1±0.1	82.5±0.2	85.4±0.2	64.1±0.0	68.4±0.0	72.4±0.0	92.8±0.0	0.528±0.008
	KNN [58]	72.2±0.0	79.0±0.1	83.2±0.2	87.5±0.2	62.3±0.1	67.1±0.0	70.2±0.1	90.1±0.2	0.421±0.009
	Decision Trees [195]	80.3±0.0	89.3±0.1	85.3±0.2	89.8±0.1	71.3±0.0	78.7±0.0	79.1±0.0	95.0±0.0	0.404±0.007
	Random Forest [196]	82.1±0.2	90.0±0.2	86.1±0.2	91.7±0.2	71.9±0.0	79.7±0.0	78.1±0.1	96.1±0.0	0.272±0.006
	XGBoost [46]	83.5±0.2	92.2±0.0	87.3±0.2	92.8±0.1	77.6±0.0	85.9±0.0	**97.3±0.0**	**99.9±0.0**	0.206±0.005
	LightGBM [70]	83.5±0.1	92.3±0.0	**87.4±0.2**	**92.9±0.1**	77.1±0.0	85.5±0.0	93.5±0.0	99.7±0.0	**0.195±0.005**
	CatBoost [71]	**83.6±0.3**	**92.4±0.1**	87.2±0.2	92.8±0.1	77.5±0.0	85.8±0.0	96.4±0.0	99.8±0.0	0.196±0.004
	Model Trees [197]	82.6±0.2	91.5±0.0	85.0±0.2	90.4±0.1	69.8±0.0	76.7±0.0	-	-	0.385±0.019
Deep Learning	MLP [198]	73.2±0.3	80.3±0.1	84.8±0.1	90.3±0.2	77.1±0.0	85.6±0.0	91.0±0.4	76.1±3.0	0.263±0.008
	VIME [79]	72.7±0.0	79.2±0.0	84.8±0.2	90.5±0.2	76.9±0.2	85.5±0.1	90.9±0.1	82.9±0.7	0.275±0.007
	DeepFM [15]	73.6±0.2	80.4±0.1	86.1±0.2	91.7±0.1	76.9±0.0	83.4±0.0	-	-	0.260±0.006
	DeepGBM [62]	78.0±0.4	84.1±0.1	84.6±0.3	90.8±0.1	74.5±0.0	83.0±0.0	-	-	0.856±0.065
	NODE [7]	79.8±0.2	87.5±0.2	85.6±0.3	91.1±0.2	76.9±0.1	85.4±0.1	89.9±0.1	98.7±0.0	0.276±0.005
	NAM [85]	73.3±0.1	80.7±0.3	83.4±0.1	86.6±0.1	53.9±0.6	55.0±1.2	-	-	0.725±0.022
	Net-DNF [50]	82.6±0.4	91.5±0.2	85.7±0.2	91.3±0.1	76.6±0.1	85.1±0.1	94.2±0.1	99.1±0.0	-
	TabNet [6]	81.0±0.1	90.0±0.1	85.4±0.2	91.1±0.1	76.5±1.3	84.9±1.4	93.1±0.2	99.4±0.0	0.346±0.007
	TabTransformer [90]	73.3±0.1	80.1±0.2	85.2±0.2	90.6±0.2	73.8±0.0	81.9±0.0	76.5±0.3	72.9±2.3	0.451±0.014
	SAINT [9]	82.1±0.3	90.7±0.2	86.1±0.3	91.6±0.2	**79.8±0.0**	**88.3±0.0**	96.3±0.1	99.8±0.0	0.226±0.004
	RLN [63]	73.2±0.4	80.1±0.4	81.0±1.6	75.9±8.2	71.8±0.2	79.4±0.2	77.2±1.5	92.0±0.9	0.348±0.013
	STG [93]	73.1±0.1	80.0±0.1	85.4±0.1	90.9±0.1	73.9±0.1	81.9±0.1	81.8±0.3	96.2±0.0	0.285±0.006

Source: https://ieeexplore.ieee.org/abstract/document/9998482

Tabular Data Generation

Why do we need tabular data generation?

- **Solves class imbalance:** Generates synthetic samples for underrepresented categories.
- **Privacy-preserving ML:** Generates realistic data without exposing sensitive information.

Popular Generative Approaches

- **GAN-based (Generative Adversarial Networks)**
  - CTGAN: Mode-specific normalization to handle categorical data.
  - PATE-GAN: Differentially private data generation.
- **VAE-based (Variational Autoencoder)**
  - TVAE: Uses probabilistic modeling to generate synthetic tabular data.

Source: https://ieeexplore.ieee.org/abstract/document/9998482

Tabular Data Generation

- **Use Cases:**
  - Finance: Synthetic credit card transaction data.
  - Healthcare: Generating anonymized patient records.
  - Cybersecurity: Simulated attack data for intrusion detection models.

Source: https://ieeexplore.ieee.org/abstract/document/9998482

Future Directions & Takeaways

- **Key Takeaways:**
  - GBDT models (XGBoost, LightGBM) still outperform deep learning in most cases, but deep learning may gain an edge as datasets scale.
  - Deep learning must improve in feature representation, interpretability, and efficiency to compete with GBDT.
  - Hybrid models (GBDT + DNN) and Transformer-based methods (SAINT, TabTransformer) show promise.
  - Tabular data generation (CTGAN, TVAE) is an emerging research direction.
- **For further research:**
  - If you want explainability & efficiency → Focus on GBDT.
  - If you are interested in deep learning for structured data → Explore hybrid models & Transformers.
  - If data is limited → Consider generative models (GAN, VAE) for augmentation.

Source: https://ieeexplore.ieee.org/abstract/document/9998482

Homework 2

- Deadline: 2025/03/27 23:59
- Lab 2
- **GitHub**: Create a “HW2" folder in your repository, "NTHU_2025_DLIA_HW", containing “HW2.ipynb" and “HW2.pdf". Ensure that you run your code, and all outputs are saved within the .ipynb files.
- **EEclass**: You are required to submit only the GitHub link of your Homework 2. Do not upload files directly to EEclass.
- **Important**: Make sure your commit is timestamped before the deadline. Late submissions might not be graded or could incur a penalty. Only the GitHub link is required on NTHU EEclass.

Reference

- Borisov, V., Leemann, T., Seßler, K., Haug, J., Pawelczyk, M., & Kasneci, G. (2022). Deep neural networks and tabular data: A survey. IEEE transactions on neural networks and learning systems.
- Ucar, T., Hajiramezanali, E., & Edwards, L. (2021). Subtab: Subsetting features of tabular data for self-supervised representation learning. Advances in Neural Information Processing Systems, 34, 18853-18865.
- Arik, S. Ö., & Pfister, T. (2021, May). Tabnet: Attentive interpretable tabular learning. In Proceedings of the AAAI conference on artificial intelligence (Vol. 35, No. 8, pp. 6679-6687).
- Popov, S., Morozov, S., & Babenko, A. (2019). Neural oblivious decision ensembles for deep learning on tabular data. arXiv preprint arXiv:1909.06312.
- Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794).
- Jordon, J., Yoon, J., & Van Der Schaar, M. (2018, September). PATE-GAN: Generating synthetic data with differential privacy guarantees. In International conference on learning representations.
- Ke, G., Xu, Z., Zhang, J., Bian, J., & Liu, T. Y. (2019, July). DeepGBM: A deep learning framework distilled by GBDT for online prediction tasks. In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 384-394).

Reference

- Somepalli, G., Goldblum, M., Schwarzschild, A., Bruss, C. B., & Goldstein, T. (2021). Saint: Improved neural networks for tabular data via row attention and contrastive pre-training. arXiv preprint arXiv:2106.01342.
- Shwartz-Ziv, R., & Armon, A. (2022). Tabular data: Deep learning is not all you need. Information Fusion, 81, 84-90.
- Yin, P., Neubig, G., Yih, W. T., & Riedel, S. (2020). TaBERT: Pretraining for joint understanding of textual and tabular data. arXiv preprint arXiv:2005.08314.

Coding Time!!