Comparing deep learning and traditional machine learning approaches for tabular data: TabNet vs XGBoost

Karla Kijac[1], Ante Ćubela[2]
[1] University of Zagreb, Faculty of Electrical Engineering and Computing, Unska ul. 3, 10000 Zagreb, Croatia
[2]University of Zagreb, Faculty of Science, Bijenička ul. 30, 10000 Zagreb, Croatia
email: karla.kijac@fer.hr, antcube.math@pmf.hr

***Abstract*— The most widespread form of data are heterogeneous tabular data, which are crucial for many computationally demanding applications. Deep neural networks have frequently shown very good performance on homogeneous datasets and have therefore been widely accepted. However, the research progress on competitive deep learning models for tabular data is stagnating since they cannot be easily adapted to tabular data for inference or data generation tasks. In this work we provide an empirical comparison of traditional machine learning method and deep learning approach across 4 real-world tabular data sets of different sizes and with different learning objectives. This knowledge is then applied to 2 new tabular datasets that model the quality of water in Kaštela Bay, provided by Ericsson. Our results indicate that algorithm based on gradient-boosted tree ensembles still mostly outperforms deep learning approach on tabular data.**

***Index Terms*—machine learning, deep learning, tabular data.**

I. Introduction

Nowdays, AI has become an integral part of the modern world. The way things function today are very much thanks to the AI that makes incredible predictions based on the enormous amount of data. These predictions help big companies earn more money, they also help us find things we search for more efficient, they help us know when to carry an umbrella, sometimes, they even save lives. To be able to do all those operations, one needs to ensure that they are precise and efficient.

The most widely spread form of data is tabular data. In this research we use heterogenous datasets from different fields of life and try to find best algorithms to make correct predictions. Not even experts in the field of AI are sure when it comes to choosing the best model for the data. For example, two research groups [2] and [1] tested machine learning and deep learning algorithms on various datasets, and each advocated for different approach. In [1] deep learning algorithm TabNet has been promoted, while in [2] experts took side with machine learning algorithm XGBoost, and each research group claimed that algorithm they chose outperforms the rest.

It is known that machine learning algorithms perform very well on tabular datasets and XGBoost, as the leading machine learning algorithm, wins numerous competitions in that field. However, the reason why to consider deep learning algorithms is that the TabNet really shook the ground when it came out claiming it outperforms machine learning algorithms. Moreover, deep learning approach including 'self-supervised' methods, no data preprocessing, efficient and iterative training does sound really appealing. The question aroused whether those complex algorithms really can outperform the old master. This was our motivation to conduct a research and compare the performance of machine learning and deep learning algorithms on existing tabular data, and apply our knowledge on a new dataset about bathing water quality.

II. Related Work

During the last decade, traditional machine learning methods, such as gradient-boosted decision trees (GBDT)[3], still showed superior performance over deep learning in the field of tabular data. Namely, there are many challenges and obstacles with deep neural networks in application to tabular data, such as mixed feature types, lack of locality, data sparsity and lack of prior knowledge about dataset structure. Furthermore, deep neural networks are considered as “black box” approach since there is no transparency or interpretability of how input data are transformed into model outputs. Therefore, tree-ensemble algorithms, such as XGBoost, are considered the best option for real-life tabular data problems [3].

In recent years, various supervised, self-supervised, and semi-supervised deep learning approaches have been proposed that explicitly address the issue of tabular data modelling [16]. More specifically, recently different deep learning methods for tabular datasets have been developed [1][7][8], and some of

them declare to outperform classical machine learning methods, such as Gradient-Boosted Decision Trees[3]. For example, TabNet[1] is a deep learning end-to-end model, that includes an encoder, in which sequential decision steps encode features using sparse learned masks and select relevant features for each row using the mask. By using sparsemax layers, the encoder forces the selection of a small set of features. Furthermore, Neural Oblivious Decision Ensembles (NODE)[8] network contains equal-depth oblivious decision trees, which are differentiable such that error gradients can backpropagate through them. Like classical decision trees, ODTs split data according to selected features and compare each with learned threshold. However, only one feature is chosen at each level, resulting in a balanced ODT that can be differentiated. Thus, the complete model provides an ensemble of differentiable trees. Finally, DNF-Net [7] simulates disjunctive normal formulas (DNF) in deep neural networks. Hard Boolean formulas are replaced with soft, differentiable versions of them. A key feature of this model is the disjunctive normal neural form (DNNF) block, which contains a fully connected layer, and a DNNF layer formed by a soft version of binary conjunctions over literals. The complete model is an ensemble of DNNFs.

Several studies have been reported that compared various deep learning models with tabular data. In [5] a large number of state-of-the-art deep learning approaches for tabular data on a wide range of datasets has been evaluated. In [2] a several different deep models for tabular data and gradient boosting decision tree algorithms have been studied regarding accuracy, training effort, and hyperparameter optimization time. It was observed that deep models had the best results on particular datasets, but not one single deep model could outperform all the others in general. This led to conclusion that efficient tabular data modelling using deep neural networks is still an open research problem.

III. Dataset Description

We used five datasets for our project. Four of them were from [2] and one from [9]. Here is a short description of each dataset. It contains information about the size of a dataset, about source of the dataset, and little something about the meaning of their features.

A. *Gas Concentration*

This dataset comes from OpenML [12], an open platform for sharing datasets, algorithms and experiments. It is one of the three datasets from [2]. It was donated by researchers from University of California San Diego. The full name of the dataset is ‘Gas drift different concentrations’ and it contains 13.9k measurements from 16 chemical sensors exposed to 6 different gases at different concentration levels. Therefore, the dataset has 6 classes, for each of the following gases: Ammonia, Acetaldehyde, Acetone, Ethylene, Ethanol and Toluene. It has 129 features, all numeric, which makes it the most complicated, and the largest classification dataset.

B. *Eye Movements*

This dataset also comes from OpenML [13] and was used in the original paper [2]. This dataset was presented by Helsinki University of Technology. It contains a pre-computed feature vectors for each word in the eye movement trajectory and 3 labels: R – relevant, I – irrelevant, C – correct. The dataset was presented for the Challenge as part of the EU Network of Excellence PASCAL Challenge Program. The Challenge was to predict from eye movement data whether a reader finds a text relevant. The dataset contains 10.9k samples with 26 features, of which 24 are numeric. The labels were encoded so that '0' represents irrelevant, '1' relevant and '2' correct.

C. *Gesture Phase*

The third dataset from [2] and OpenML [11], also a classification problem. The dataset was donated by University of Sao Paolo – Brazil. It contains 32 different features that represent velocity and acceleration of hands and wrists extracted from 7 videos of people gesticulating. The aim was to predict in which phase the gesture was. The phases were divided into six: D – rest position (from Portuguese "descanso"), P – preparation, S – stroke, H – hold, R – retraction, meaning there are 6 classes in this dataset. This dataset has 9.8k samples which makes it the smallest in the classification group.

D. *Year Prediction*

The fourth, and last, dataset from [2] is a regression problem dataset. This dataset was huge for our terms. It has 515k samples and 90 features. It was downloaded from the UC San Diego web page that donated this dataset for Kaggle challenge The Million Song Dataset. The features in dataset are different measures taken on songs and the last feature is integer that should be predicted and represents the year from which the song is.

E. *WQ Kaštela*

Very recent dataset on bathing water quality measured in Kaštela Bay provided by Ericsson Nikola Tesla. The monitoring was conducted by Teaching Institute of Public Health of Split-Dalmatia County at 11 bathing sites alongside Kaštela Bay during 2015-2019 bathing seasons. It measures the contamination of bathing water with E. Coli and intestinal enterococci in order to be able to predict possible dangerous/unhealthy times for taking a swim. Since the bathing season is not all year long and the measurements were done fortnightly there are only 612 samples in this regression problem dataset. Prescribed by BWD, the water quality can be classified as Excellent, Good, Sufficient and Poor so the dataset is also a classification problem.

IV. THEORY

A. Tabular Data

Tabular data, or also called structured data is simply data organized in rows and columns. It is the most widely spread type of data, and one of the oldest forms of keeping data for statistical analysis. Unlike unstructured data, which are mostly images, video and audio files, JSON files etc., structured data is not homogenous and usually contains more types of data. They can contain numeric (such as currency amounts, time duration, temperature, velocity) and non-numeric (such as strings, objects) values. The variety of data types makes them harder to interpret.

B. XGBoost

Extreme gradient boosting decision tree machine learning algorithm, shortly XGBoost [3] is currently leading machine learning library for regression, classification, and ranking problems [17]. Ensemble learning, where tree boosting is a common technique, is a type of machine learning that enlists many models to make predictions together. Boosting algorithms are distinguished from other ensemble learning techniques by building a sequence of initially weak models into increasingly more powerful models. Gradient boosting algorithms choose how to build a more powerful model using the gradient of a loss function that captures the performance of a model.

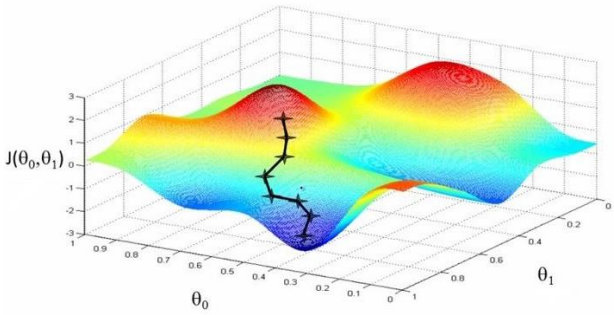


Fig. 1. Gradient descent on cost function

The gradient descent works like a compass, but instead of North it "shows" you where to go to get to the lowest point efficiently, that is to the minimum value of the cost function.

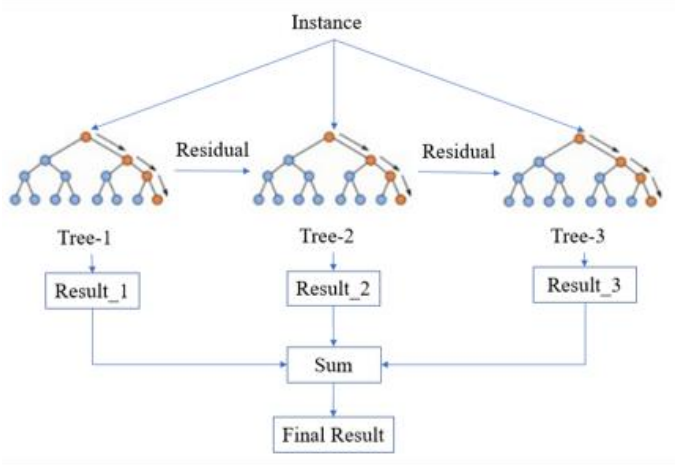
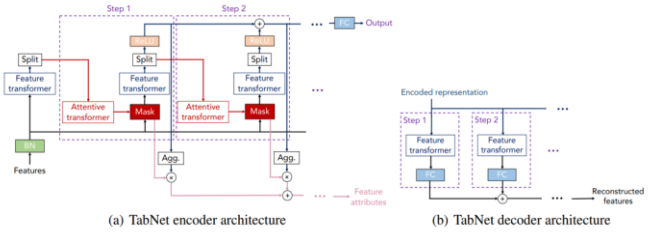


Fig. 2. XGBoost architecture

XGBoost achieves state-of-the-art results on many tabular datasets because of its architecture Figure 2. It can be seen that new models are being created from previous models' residuals and combined to make the final prediction.

C. TabNet

TabNet [1] is a novel high-performance and interpretable canonical deep tabular data learning architecture developed by Google. It is one of the first transformer-based models for tabular data. Like a decision tree, the TabNet architecture comprises multiple subnetworks that are processed in a sequential hierarchical manner. Each subnetwork corresponds to one decision step. In each decision step it uses feature masks to select features that it will focus on to save valuable resources. The feature mask of a decision step is trained using attentive information from the previous step. Here, a feature transformer module decides which features should be passed to the next decision step and which should be used to obtain the output at the current decision step. By using sequential attention to choose which features to reason, TabNet is one of the few deep neural networks that offers different layers of interpretability and more efficient.



(a) TabNet encoder architecture (b) TabNet decoder architecture

Fig. 3. TabNet architecture

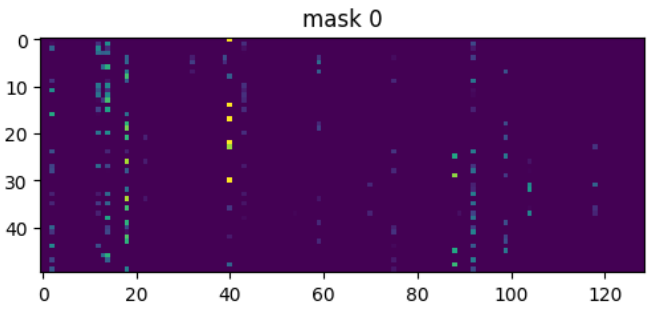


Fig. 4. Most important features through mask 0 on Gas Concentration example

V. Methodology

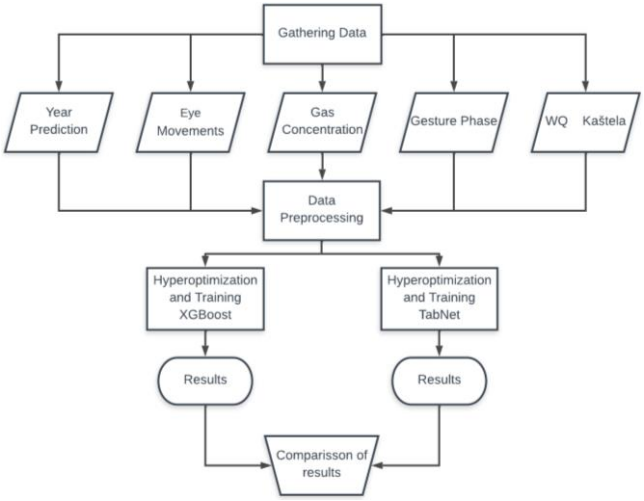


Fig. 5. Information flow

A. Data Preparation

The 4 datasets from [2] were chosen because they do not require any preprocessing. The train/test split for Year Prediction was provided by the authors. We used 20% of training data for the validation set, chosen randomly. For the other 3 datasets from the paper, we used stratified split to make 5 partitions where the train/test/vali ratio was 70%/20%/10%, as was used in the paper. WQ Kaštela dataset did not required any preprocessing as well, and was provided with a train/test split. We used stratified split to make 5 validation sets from the training data, so that the train/vali ratio was 80%/20%. The training sets were standardized to have a mean of 0 and a standard deviation of 1, and the other sets were normalized with the statistics from the training set.

B. Hyperoptimization And Training

For the hyperoptimization process, we used HyperOpt package [4] in combination with HpSklearn and Sklearn package. We used *fmin()* function from the HyperOpt package with Bayesian optimization with the Tree-structured Parzen Estimator algorithm. There were 10 hyperparameters being optimized for the XGBoost model, and 7 for the TabNet model. The exact search spaces are listed below. The search space for XGBoost is:

- Number of estimators: Uniform distribution [100,4000]
- Learning rate: Log-Uniform distribution $[e^{-7}, 1]$
- Max depth: Discrete Uniform distribution [1,10]
- Subsample: Uniform distribution [0.2,1]
- Colsample by tree: Uniform distribution [0.2,1]
- Colsample by level: Uniform distribution [0.2,1]
- Min child weight: Log-Uniform distribution $[e^{-16}, e^{5}]$
- Alpha: Uniform choice {0, Log-Uniform distribution $[e^{-16}, e^{2}]$}
- Lambda: Uniform choice {0, Log-Uniform distribution $[e^{-16}, e^{2}]$}
- Gamma: Uniform choice {0, Log-Uniform distribution $[e^{-16}, e^{2}]$}.

The search space for TabNet is:

- Learning rate: Uniform distribution $[e^{-5}, 1]$
- Decision layer dim: Discrete Uniform distribution [20,60]
- Attention embedding dim: Discrete Uniform distribution [20,60]
- Number of steps: Discrete Uniform distribution [2,10]
- Batch momentum: Uniform distribution [0.5,0.98]
- Relaxation factor: Uniform distribution [1,2]
- Batch size: Uniform distribution {512,1024,2048,4096,8192}

For the WQ Kaštela dataset, the batch size was uniformly chosen from the set {16,32,64,128,256}.

We ran 500 trials of hyperoptimization with the same random seed for all the datasets and models. TabNet model performed 200 epochs with early stopping after 10 epochs without improvements on the WQ Kaštela dataset, and 30 epochs without improvements on other datasets. No early stopping was used on Year Prediction dataset and only 50 trials were performed for each algorithm (due to the restrictions in computational power). Adam optimizer was used for the TabNet model. The set of hyperparameters with the best loss on the validation set after 500 trials was chosen as optimal. Best loss was chosen as the minimum cross-entropy loss or mean squared error, for classification or regression problems, respectively. The two metrics are generally considered to be a great choice for comparing the performance of different algorithms.

VI. Discussion and Results

We report the average cross-entropy loss/mean squared error from 3 to 5 partitions of the datasets, measured on the test sets.

	Eye Movements	Gas Concentrations	Gesture Phase
XGBoost	54.15	2.26	79.81
TabNet	87.1	3.38	117.7

Fig. 6. Comparison of the performance through cross entropy loss (×100) of the two algorithms on classification datasets

	WQ ECOLI	WQ CE	Year Prediction
XGBoost	106.33	82.21	82.73
TabNet	132.19	92.48	78.87

Fig. 7. Comparison of the performance through mean squared error of the two algorithms on regression datasets

The results for XGBoost algorithm were on par with the results from original papers [2][7], while TabNet did not reach close to the reported performance. Results for the outcome are differences in implementation of TabNet and the hyperoptimization process. TabNet did reach the performance reported in the paper [2] at time, but the performance was far from consistent. The model was quite prone to overfitting and did not generalize well. Reaching very good loss metrics on the validation set often did not reflect in a good loss on the test set. We managed to improve the stability by adjusting the parameter space from the paper, motivated with [15].

The process of hyperoptimization was much more difficult for TabNet as well. There is no good measure to compare the complexity of the hyperoptimizations, but the hyperoptimization of TabNet was much more expensive in both the time and the computing power required. The results of both model performances with their default parameters show that the defaults parameters for XGBoost are far more robust than for TabNet.

TabNet did outperform XGBoost on the dataset Year Prediction, but due to our computing restrictions, the hyperoptimization process is not as representative. Although, that result is in line with many empirical results researchers encountered for deep learning models. Deep learning models often outperform state-of-the-art machine learning models on big datasets (about 500k samples or more).

The results measured on the two WQ Kaštela regression problems align with the other results yielded. Due to a small size of the dataset (473 samples in the trainings sets), the process of hyperoptimization was reasonably fast. TabNet reported a better loss in few partitions, but the results deviated a lot more. The problem with generalization of the model on the test set was highlighted, and its tendency to overfit.

VII. CONCLUSION

In this paper, a set of performance and scalability benchmarks were performed in order to compare the machine learning model XGBoost with deep learning model TabNet. The results show that XGBoost outperforms TabNet in all performed benchmarks by a margin. TabNet model does not generalize very well and often overfits on the small datasets, which is especially highlighted on the WQ dataset, where the results were very inconsistent. The results of this paper indicate that XGBoost still remains state-of-the-art algorithm when working with tabular datasets.

ACKNOWLEDGMENT

Authors would like to thank Marija Todorić for guidance and help, also Ericsson for providing us the equipment and work space. Also, we are grateful to Toni Mastelić that provided us the WQ Kaštela dataset.

REFERENCES

[1] S. O. Arik and T. Pfister, "TabNet: Attentive interpretable tabular learning," arxiv:1908.07442, 2019.
[2] R. Shwartz-Ziv and A. Armon, "Tabular Data: Deep Learning is Not All You Need," arXiv preprint arXiv:2106.03253, 2021.
[3] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2016, pp. 785-794.
[4] James Bergstra, Brent Komer, Chris Eliasmith, Dan Yamins, and David D Cox. Hyperopt: a python library for model selection and hyperparameter optimization. Computational Science & Discovery, 8(1):014008, 2015
[5] Y. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko, "Revisiting deep learning models for tabular data," arXiv preprint arXiv:2106.11959, 2021.
[6] Liran Katzir, Gal Elidan, and Ran El-Yaniv. Net-dnf: Effective deep modeling of tabular data. In 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net, 2021. E. P. Wigner, "Theory of traveling-wave optical laser," *Phys. Rev.*, vol. 134, pp. A635–A646, Dec. 1965.
[7] Ami Abutbul, Gal Elidan, Liran Katzir, and Ran El-Yaniv. Dnf-net: A neural architecture for tabular data. arXiv preprint arXiv:2006.06465, 2020.
[8] Sergei Popov, Stanislav Morozov, and Artem Babenko. Neural oblivious decision ensembles for deep learning on tabular data. In 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net, 2020.
[9] Džal D.; Nižetić Kosović, I.; Mastelić. T.; Ivanković, D.; Puljak, T.; Jozić, S. Modelling Bathing Water Quality Using Official Monitoring Data. Water 2021, 13, 3005.
[10] https://dreamquark-ai.github.io/tabnet/
[11] https://www.openml.org/search?type=data&sort=runs&id=4538&status=active, Gesture Phase dataset
[12] https://www.openml.org/search?type=data&sort=runs&id=1477&status=active, Gas Concentrations dataset
[13] https://www.openml.org/search?type=data&sort=runs&id=1044&status=active, Eye Movements dataset
[14] https://archive.ics.uci.edu/ml/datasets/yearpredictionmsd,Year Prediction dataset
[15] S. Thapa, Implementing TabNet in PyTorch, site: https://towardsdatascience.com/implementing-tabnet-in-pytorch-fc977c383279
[16] V. Borisov, T. Leemann, S. Kathrin, J. Haug, M. Pawelczyk, G. Kasneci, "Deep Neural Networks and Tabular Data: A Survey"
[17] What is XGBoost?, site: https://www.nvidia.com/en-us/glossary/data-science/xgboost/

K. Kijac was born in Zagreb, 2001. She went into V. Gymnasium where she fell in love with ICT. Currently she is a student at Faculty of Electrical Engineering and Computing at University of Zagreb. She loves sport and being outdoors. Trains karate, and rides horses, and loves to spend time in nature. Besides nature, and animals she loves to get lost in books. This summer she participated in Ericsson Nikola Tesla Summer Camp. Her fields of interests are: communication technologies and networks, IOT and AI.

A. Ćubela was born in Mostar in 1998. He attended fra Dominik Mandić gymnasium in Široki Brijeg. Afterwards he received Bachelor's degree in Mathematics on the Faculty of Science in the University of Zagreb. He is currently pursuing the Master's degree in Applied Mathematics. He has participated in Ericsson Nikola Tesla Summer Camp in Zagreb in 2022. His fields of interest are: numerical simulations, machine learning and game development.

In his free time he enjoys playing and watching sports, and playing board and video games.