Minimal Supervision, Maximum Accuracy: TabPFN for Microcontroller Performance Prediction

Minimal Supervision, Maximum Accuracy: TabPFN for Microcontroller Performance Prediction

Nicolò Bellarmino*, Riccardo Cantoro*, Martin Huch†, Tobias Kilian†

***Abstract*—Microcontroller (MCU) performance screening ensures devices meet the maximum operating frequency $F_{\max}$ specification. Speed Monitors (SMONs), implemented as ring oscillators, are used to estimate $F_{\max}$. Traditional machine learning (ML) models have been explored for this task but require extensive feature engineering and tuning. This work investigates Tabular Foundation Models, specifically TabPFN, for MCU performance prediction. TabPFN leverages in-context learning, enabling accurate inference without dataset-specific training. We evaluate its performance on a composite dataset combining four distinct MCU product families. Results show that TabPFN matches or exceeds baseline ML models while eliminating the need for manual optimization, offering a promising direction for efficient screening in semiconductor manufacturing with minimal human supervision**

***Index Terms*—Fmax, Speed Monitors, Ring Oscillators, Speed Binning, Machine Learning, Device Testing, Manufacturing, Tabular Foundation Models**

I. Introduction

Microcontroller (MCU) performance screening[1] is a crucial step in semiconductor manufacturing, aimed at identifying devices that fail to meet operational specifications, particularly the maximum operating frequency ($F_{\max}$). Machine Learning (ML) has emerged as a promising approach for predicting $F_{\max}$ by leveraging indirect, process-sensitive measurements. Among these, on-chip ring oscillators—commonly known as Speed Monitors (SMONs)—provide frequency readings strongly correlated with device performance.

Traditional ML methods, including ensemble models such as XGBoost and CatBoost [2], have demonstrated strong predictive capabilities in this context. However, deploying these models in industrial settings remains challenging due to their reliance on domain expertise, extensive feature engineering, and careful hyperparameter tuning. This limits their scalability across different product families and slows down the development cycle.

Recent advancements in deep learning for tabular data have introduced *Tabular Foundation Models* (TabFMs), which aim to offer robust generalization across tasks by pre-training on synthetic datasets. Notably, models like TabPFN [3] utilize *In-Context Learning* (ICL) to make predictions conditioned on a small set of labeled examples. Inspired by large language models, this approach allows the model to perform inference without requiring task-specific training.

* Politecnico di Torino (Turin, Italy). † Infineon Technologies AG (Munich, Germany). Authors are listed in alphabetical order.

TabPFN is pre-trained to approximate Bayesian inference over a wide range of classification tasks and produces predictions through a single forward pass. It eliminates the need for hyperparameter tuning and manual feature selection, making it suitable for rapid deployment with minimal human supervision. This is particularly advantageous in the semiconductor domain, where product heterogeneity and evolving test conditions demand adaptable and low-maintenance solutions.

In this work, we explore the potential of TabPFN for MCU performance screening. We evaluate its predictive accuracy and generalization capabilities on datasets from multiple MCU product families and compare it to established ML baselines. Our results demonstrate that TabPFN delivers competitive performance with significantly reduced development overhead, suggesting that foundation models can serve as a practical alternative to conventional ML pipelines in industrial testing applications.

The remainder of this paper is organized as follows: Section II discusses relevant background and related work. Section III presents our experimental setup and results. Finally, Section IV concludes the paper and outlines future research directions.

II. Background

A. MCU Performance Screening and Speed Monitors

Microcontroller (MCU) performance screening aims to assess whether devices meet target specifications, such as the maximum operating frequency ($F_{\max}$). This task is traditionally supported by characterization procedures that establish correlations between measurable device features and $F_{\max}$ [4]–[10].

A widely adopted technique employs *Speed Monitors* (SMONs), implemented as on-chip ring oscillators (ROs), to serve as proxies for performance estimation [1], [11]–[14]. These ROs are embedded at various physical locations on the die to capture spatial process variations. SMONs are grouped into modules, each containing a heterogeneous set of ROs [13], and multiple identical modules are distributed across the chip to enhance observability. The frequency responses measured from these structures across multiple production lots constitute the input features for machine learning (ML)-based prediction models.

B. Limitations of Traditional Machine Learning Models

Traditional ML models, especially tree-based methods such as XGBoost and CatBoost [2], have demonstrated effective

predictive performance in estimating $F_{\max}$. However, these models require dataset-specific training procedures and manual tuning of hyperparameters, often coupled with substantial feature engineering. Moreover, their applicability is limited by assumptions of training and inference distribution alignment, leading to poor generalization across different MCU product families or manufacturing batches. Each deployment typically involves a time-intensive model development cycle, making it challenging to scale across a broad range of devices or process nodes.

C. In-Context Learning and Tabular Foundation Models

To address the limitations of traditional supervised learning, recent research has explored *In-Context Learning* (ICL), a paradigm in which models learn to perform new tasks by conditioning on a small number of labelled examples provided at inference time [3], [15], [16]. In contrast to conventional approaches, ICL does not require any gradient-based training on task-specific data. Instead, the model leverages patterns and priors learned during pretraining to infer directly from the context provided during evaluation.

ICL first emerged in the field of natural language processing, where LLMs such as GPT-3 [17] demonstrated the ability to perform few-shot or zero-shot learning by processing prompts containing task examples. This prompted the development of analogous strategies for structured data, giving rise to *Tabular Foundation Models* (TabFMs). These models are pretrained on large corpora of synthetically generated tabular datasets that represent a wide range of causal structures, feature distributions, and learning tasks. As a result, TabFMs can generalize to unseen real-world datasets with little or no adaptation, removing the need for task-specific retraining or extensive feature manipulation.

D. TabPFN: A Transformer-Based Model for Tabular Data

One prominent example of a Tabular Foundation Model is the *Tabular Prior-Data Fitted Network* (TabPFN) [3], which uses a transformer architecture to implement ICL for tabular regression and classification tasks. During pretraining, TabPFN is exposed to millions of synthetically generated datasets with diverse structures and statistical properties. The model learns a strong Bayesian prior over this distribution of tasks, enabling it to reason probabilistically about new examples during inference.

At test time, TabPFN takes as input a small set of labelled examples and one or more unlabelled instances, and produces predictions through a single forward pass, without any model parameter updates. TabPFN's ICL-based inference enables it to adapt to new datasets with only a handful of labelled samples, making it especially attractive in low-data regimes or high-variability environments. Importantly, the model maintains competitive accuracy while drastically reducing the need for hyperparameter tuning and feature engineering.

Given the diversity of products and variability inherent to semiconductor manufacturing, these properties make TabPFN particularly suitable for MCU performance screening. Its ability to generalize across different distributions and to operate effectively with minimal labelled data aligns well with the operational constraints of industrial screening workflows.

III. Discussion

This study investigates the application of *TabPFN* in the context of MCU performance screening, focusing on its ability to generalize across heterogeneous datasets derived from multiple product families. Our goal is to assess whether TabPFN can provide a competitive, yet more accessible and easy-to-use alternative to traditional ML models commonly used in semiconductor manufacturing pipelines.

A. Composite Dataset Construction

To evaluate the generalization capabilities of TabPFN, we constructed a composite dataset by merging samples from four distinct MCU product families, denoted as A, B, C, and D. This integration introduces substantial variation in both feature distributions and performance labels, creating a realistic benchmark that reflects the complexity of industrial manufacturing environments. Each product differs in terms of architectural design, number and distribution of SMON modules, and target performance classes, thereby enabling a comprehensive evaluation of model robustness.

Product A consists of 1,032 labeled samples obtained from a high-performance MCU family. The device integrates six SMON modules, each comprising approximately 130 SMONs strategically distributed across the die. This dense monitoring infrastructure enables detailed profiling of intra-die process variability and timing criticality.

Product B includes 1,156 samples from a medium-performance MCU variant, featuring five SMON modules. The product features five SMON modules with architectural similarities to those in Product A but adapted to different performance envelopes.

Product C contributes 1,642 samples from a different high-performance design family. It integrates four SMON modules and reflects broader manufacturing and architectural differences, offering complementary variation for assessing model adaptability.

Product D provides 412 samples and represents a low-complexity design with a single SMON module. Its reduced spatial observability and simplified structure yield a markedly different feature distribution. Product D serves as a critical test case for evaluating model robustness under limited feature representation and distributional shift.

B. Feature Harmonization Strategies

Due to architectural differences across the MCU products, aligning their feature spaces required a harmonization strategy. We adopted a feature intersection approach, retaining only the SMONs common to all four products. This reduced the dimensionality to approximately 128 features while preserving physical interpretability and enabling consistent cross-product comparisons.

Figure 1 presents a 3D visualization of the unified dataset using Principal Component Analysis (PCA) [18]. The figure shows the projection of each sample onto the first two principal components—capturing the largest variance in the data—while the vertical axis reflects the normalized $F_{\max}$ values.

The PCA plot reveals notable patterns across products. Product A and Product C exhibit substantial overlap, suggesting structural and statistical similarities in their SMON-derived features. In contrast, Product B, despite being architecturally related to Product A, displays a more distinct distribution. This divergence likely arises from process and layout variations, offering a valuable case study for assessing the robustness of predictive models under moderate domain shifts.

C. Baseline Models and Evaluation

We compared TabPFN to a suite of well-established baseline models. These include tree-based algorithms such as *Random Forest* and *XGBoost* [2], [19], as well as linear models including *Ridge Regression* and *Polynomial Ridge Regression* [14], [20], [21]. All models were optimized using 5-fold cross-validation.

For the tree-based models, we adopted a *Random Search* strategy across 400 candidate hyperparameter configurations. For Ridge-based models, we performed an exhaustive search across more than 10,000 values of the regularization parameter λ. The polynomial variant applied a second-degree transformation to the input features, which is a common form of manual feature engineering for capturing non-linear interactions in tabular domains.

In contrast, *TabPFN* was deployed without any form of feature preprocessing or model-specific tuning. The model operates directly on normalized raw data and supports inference via a single forward pass without requiring training or gradient updates. This setup significantly reduces the time and expertise needed for model development.

Each model was evaluated using five different train-test splits, with an 80% training and 20% testing proportion.

D. Results and Analysis

The results of the composite dataset experiments are reported in Figure 2 and Table I, in which we show results of the whole dataset (*Overall*) and grouped by product. *TabPFN* consistently achieved competitive or superior performance in terms of R^2 score, normalized Root Mean Square Error (nRMSE), normalized Mean Absolute Error (nMAE). Notably, despite the variability in data distribution and SMON architecture, TabPFN preserved high predictive accuracy across all MCU families. In particular, for each MCU product, the boxplots in Figure 2 highlight the consistently lower prediction variability of *TabPFN* compared to traditional models. This reduced spread indicates that *TabPFN* not only achieves high average accuracy but also delivers more stable and reliable predictions across different data splits, underscoring its robustness in handling process and architectural variations.

These results suggest that TabPFN's ability to internalize a rich prior during pretraining—paired with its capacity to adapt through in-context inference—makes it particularly well-suited to applications characterized by distribution shifts, small data regimes, and architectural heterogeneity. Furthermore, the elimination of dataset-specific optimization dramatically simplifies deployment, highlighting TabPFN as a compelling candidate for scalable, production-ready ML pipelines in semiconductor testing.

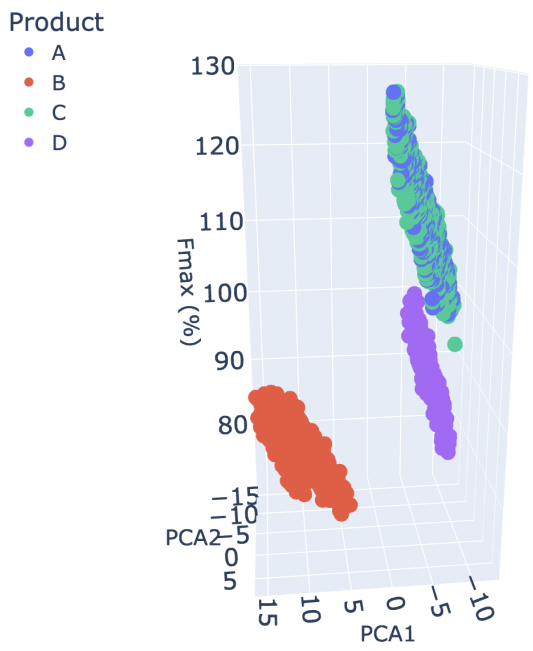

Fig. 1. Composite dataset comprising four MCU products. The relationship between SMON features and $F_{\max}$ appears approximately linear across all products. However, each product exhibits a distinct distribution, with Products B and D showing significant deviation from the distributions observed in Products A and C.

The results demonstrate that *TabPFN* serves as a ready-to-deploy alternative to conventional machine learning models, particularly in scenarios where manual feature engineering and hyperparameter tuning pose significant resource and time constraints. Its consistent performance across diverse MCU product families highlights the robustness and adaptability of Tabular Foundation Models. These characteristics position TabPFN as a promising candidate for real-world deployment in semiconductor manufacturing workflows, offering reliable predictive capabilities without the need for product-specific calibration or retraining.

IV. Conclusions

In this work, we explored the use of Tabular Foundation Models, with a focus on *TabPFN*, for microcontroller (MCU) performance screening. Our study was conducted on a composite dataset integrating data from four MCU product families, each characterized by different architectures, SMON configurations, and manufacturing variations.

The experimental results demonstrate that *TabPFN*, even without any product-specific tuning or feature preprocessing,

TABLE I
MODEL PERFORMANCE BY PRODUCT AND OVERALL. HIGHER R^2 AND LOWER NRMSE/NMAE INDICATE BETTER PERFORMANCE.

Algorithm	R^2 (%)					nRMSE (%)					nMAE (%)				
	A	B	C	D	Overall	A	B	C	D	Overall	A	B	C	D	Overall
Poly Ridge	94.09	85.05	91.85	94.20	99.19	1.14	1.77	1.28	1.59	1.37	0.90	1.36	1.00	1.26	1.07
Random Forest	95.78	83.90	93.86	94.10	99.30	0.94	1.84	1.09	1.60	1.27	0.71	1.43	0.82	1.26	0.96
Ridge	94.15	83.28	91.34	92.71	99.13	1.13	1.87	1.31	1.77	1.43	0.90	1.44	1.02	1.39	1.11
TabPFN	**97.13**	**86.73**	**96.21**	**94.31**	**99.48**	**0.79**	**1.67**	**0.87**	**1.57**	**1.10**	**0.60**	**1.27**	**0.66**	**1.24**	**0.82**
XGBoost	96.71	84.37	94.43	92.95	99.34	0.84	1.81	1.05	1.75	1.24	0.61	1.38	0.77	1.40	0.91

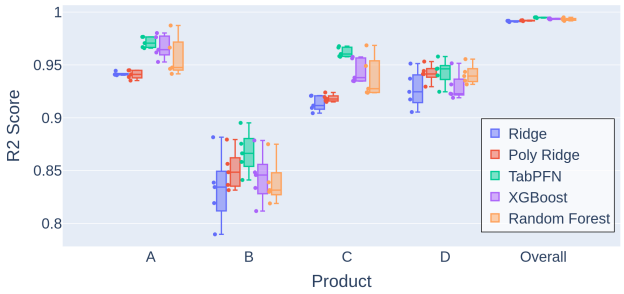

Fig. 2. BoxPlots of R2 Score on each product and overall. In general, TabPFN has lower variance and higher score.

achieves performance on par with or superior to traditional machine learning models. Notably, TabPFN required no manual feature engineering or hyperparameter optimization, drastically reducing development and deployment overhead.

These findings underscore the potential of Tabular Foundation Models as a practical alternative to conventional ML workflows, particularly in industrial settings where rapid prototyping, scalability across product lines, and minimal configuration effort are critical. The ability to leverage in-context learning for tabular data further positions TabPFN as a step toward more adaptive and self-sufficient ML systems in manufacturing and testing environments, even with minimal human supervision.

Future Directions

This study opens several promising research avenues. We plan to investigate the potential of fine-tuning TabPFN on domain-specific datasets to assess whether targeted adaptation can yield further performance improvements, especially for product families exhibiting complex or non-linear behavior.

REFERENCES

[1] N. Bellarmino *et al.*, "Exploiting active learning for microcontroller performance prediction," in *IEEE European Test Symposium (ETS)*, 2021.

[2] T. Chen *et al.*, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16, ACM, Aug. 2016.

[3] N. Hollmann *et al.*, *Tabpfn: A transformer that solves small tabular classification problems in a second*, 2023.

[4] K. von Arnim *et al.*, "An Effective Switching Current Methodology to Predict the Performance of Complex Digital Circuits," in *IEEE International Electron Devices Meeting (IEDM)*, 2007.

[5] G. Sannena *et al.*, "Low overhead warning flip-flop based on charge sharing for timing slack monitoring," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2018.

[6] T. B. Chan *et al.*, "DDRO: A novel performance monitoring methodology based on design-dependent ring oscillators," in *Thirteenth International Symposium on Quality Electronic Design (ISQED)*, May 2012.

[7] F. Angione *et al.*, "Test, reliability and functional safety trends for automotive system-on-chip," in *2022 IEEE European Test Symposium (ETS)*, 2022.

[8] P. Variyam *et al.*, "Prediction of analog performance parameters using fast transient testing," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, 2002.

[9] H.-G. Stratigopoulos *et al.*, "Error moderation in low-cost machine-learning-based analog/rf testing," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, 2008.

[10] H.-G. Stratigopoulos *et al.*, "Defect filter for alternate rf test," in *2010 15th IEEE European Test Symposium*, 2010.

[11] N. Bellarmino *et al.*, "Microcontroller Performance Screening: Optimizing the Characterization in the Presence of Anomalous and Noisy Data," in *IEEE International Symposium on On-Line Testing and Robust System (IOLTS)*, 2022.

[12] N. Bellarmino *et al.*, "Semi-Supervised Deep Learning for Microcontroller Performance Screening," in *IEEE European Test Symposium (ETS)*, 2023.

[13] N. Bellarmino *et al.*, "Feature Selection for Cost Reduction In MCU Performance Screening," in *IEEE 24th Latin American Test Symposium (LATS)*, 2023.

[14] N. Bellarmino *et al.*, "A Multi-Label Active Learning Framework for Microcontroller Performance Screening," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, 2023.

[15] B. van Breugel *et al.*, *Why tabular foundation models should be a research priority*, 2024.

[16] Q. Dong *et al.*, *A survey on in-context learning*, 2024.

[17] T. B. Brown *et al.*, *Language models are few-shot learners*, 2020.

[18] B. Schölkopf *et al.*, "Kernel principal component analysis," in *Artificial Neural Networks — ICANN'97*, W. Gerstner *et al.*, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 1997.

[19] L. Breiman, "Random forests," en, *Machine Learning*, Oct. 2001.

[20] A. E. Hoerl *et al.*, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, 1970.

[21] N. Bellarmino *et al.*, "Cosmo: Compressed sensing for models and logging optimization in mcu performance screening," *IEEE Transactions on Computers*, 2024.