ML-assisted SRAM Soft Error Rate Characterization: Opportunities and Challenges

Masanori Hashimoto
Kyoto University
Kyoto, Japan

Ryuichi Yasuda
Kyoto University
Kyoto, Japan

Kazusa Takami
Kyoto University
Kyoto, Japan

Yuibi Gomi
Kyoto University
Kyoto, Japan

Kozo Takeuchi
Japan Aerospace Exploration Agency
Tsukuba, Japan

ABSTRACT

Soft errors from cosmic rays are a significant concern for reliability-critical applications such as autonomous driving and supercomputers. In this paper, we review soft error rate (SER) estimation for SRAM, the most sensitive component in digital logic chips, and explore how machine learning can assist in SRAM SER characterization. We propose an efficient discriminator construction method for single-event upset (SEU) using active learning and adaptive hyperparameter tuning in the learning algorithm. This method iteratively labels samples through technology computer-aided design (TCAD) simulations, determining whether an upset occurs for an unlabeled sample with the lowest confidence in prediction. Our approach eliminates the need for empirical modeling based on tacit knowledge, systematically building a model while reducing the training data needed to achieve sufficient event-wise accuracy. Experiments with a 12-nm SRAM show that the training data required to achieve the same accuracy was reduced by 41% for 80% accuracy and by 31% for 85% accuracy. Finally, we discuss future directions and challenges in advanced nano-sheet and CFET transistors.

CCS CONCEPTS

• **Hardware** → **Transient errors and upsets**; • **Computing methodologies** → *Active learning settings.*

KEYWORDS

Single-Event Upset, Soft Error, TCAD, Active Learning, Machine Learning

ACM Reference Format:
Masanori Hashimoto, Ryuichi Yasuda, Kazusa Takami, Yuibi Gomi, and Kozo Takeuchi. 2025. ML-assisted SRAM Soft Error Rate Characterization: Opportunities and Challenges. In *30th Asia and South Pacific Design Automation Conference (ASPDAC '25), January 20–23, 2025, Tokyo, Japan.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3658617.3703316

1 INTRODUCTION

With the miniaturization, low-voltage operation, and high integration of VLSI, radiation effects have become a significant problem, especially for applications demanding high reliability, such as self-driving cars. In terrestrial environments, neutron-induced soft errors have emerged as a primary reliability concern, while in space applications, protons and heavier ions also pose serious risks.

To improve the reliability of these devices, evaluating the impact of soft errors has become critical. Soft errors refer to temporary malfunctions that occur when one or more bits stored in memory are flipped. This phenomenon is primarily caused by radiation-induced electron-hole pair generation within the device, leading to charge collection at the drain and causing its potential to fluctuate abnormally. A single-event upset (SEU) occurs when a bit flip is triggered by a single particle strike on a device, such as in static random access memory (SRAM). Various methods have been developed to estimate the soft error rate (SER), with most approaches based on Monte Carlo simulations [1, 11, 12, 22].

These methods perform particle transport simulations, precisely modeling particle-atom interactions, ion transportation, and charge deposition within devices using tools such as GEANT4 [3] or PHITS [17]. The most accurate estimation methods, such as [1], incorporate technology computer-aided design (TCAD) device simulations for each charge deposition event to determine whether bit upsets occur. However, estimating SER with TCAD simulations presents a challenge, as each simulation can take considerable time, often exceeding ten hours to complete. Moreover, because Monte Carlo methods rely on random sampling, a large number of trials are needed to achieve adequate statistical accuracy.

Following the Monte Carlo particle transport simulation, the sensitive volume (SV) method [20] is commonly used to determine whether bit upsets occur, due to its simplicity. The basic single sensitive volume (SSV) approach identifies a bit upset by checking if the charge deposited in the SV exceeds a defined critical charge value. However, this approach often depends on an empirically determined SV and an estimated critical charge threshold. Consequently, without empirical knowledge or measurement data, it becomes challenging to estimate SER with confidence. Furthermore, even when SER is accurately estimated, the event-wise accuracy, namely the discrimination accuracy for each charge deposition event, remains low. This issue will be discussed in detail in Section 2.

To address this problem, machine learning has been integrated with TCAD simulations [6, 7]. In this approach, TCAD simulation

ASPDAC '25, January 20–23, 2025, Tokyo, Japan

ACM ISBN 979-8-4007-0635-6/25/01
https://doi.org/10.1145/3658617.3703316

data and corresponding deposited charges across multiple sensitive volumes (MSVs) are prepared as training data, and then the discriminator is constructed. However, applying this method presents challenges, as generating sufficient training data (e.g., over 1000 samples) is difficult due to the slow speed of TCAD simulations. Furthermore, the method in [7] was validated only on 65-nm SOI devices, whereas FinFETs are currently prevalent as state-of-the-art technology, making experimental validation for FinFETs essential.

This paper proposes an SEU discriminator construction method incorporating active learning and adaptive hyperparameter tuning. Compared to standard machine learning approaches, the proposed method reduces the amount of training data required to achieve sufficient accuracy. In this study, we apply the method to 12-nm FinFET SRAM, demonstrating its superiority in accuracy over conventional SV methods and its reduced training data requirements compared to typical machine learning approaches. Experimental results show that the number of training samples needed to achieve 80% and 85% accuracy was reduced by 41% and 31%, respectively, whereas conventional SSV and MSV methods suffer from low accuracies of 57% and 68%, respectively.

2 MOTIVATION

For planar MOSFETs, the sensitive volume has traditionally been empirically allocated to the drain of the off-state NMOS, with a sub-micron depth. The charge collection efficiency, which is defined as the ratio of charge collected at the drain to the total deposited charge, varies depending on the location of the charge deposition (an example will be shown later in Fig. 7). Therefore, assigning a single sensitive volume is a significant approximation. Additionally, for the FinFET structure, there is no established consensus in the literature regarding sensitive volume allocation.

Despite the issues mentioned, the SSV method is commonly used because the SER can be reproduced by adjusting the critical charge threshold value [2]. Even with suboptimal SSV locations and sizes, fitting the critical charge value can mask these inaccuracies. In such cases, the SER may be reproduced, but the SEU occurrence for individual particle events cannot be accurately predicted. Consequently, event-wise accuracy cannot be expected with the SSV method.

Event-wise accuracy is crucial for estimating multiple cell upsets (MCUs) because a single particle can cause upsets in adjacent SRAM cells. Accurately estimating MCUs requires considering the spatial location of the particle track across multiple cells. However, the SSV method is not well suited for this purpose.

As an improvement, MSV methods are also used in the literature. To account for the spatial distribution of charge collection efficiency, multiple SVs with varying collection efficiencies are allocated. The total collected charge is then estimated as the sum of the products of the charge collection efficiency and the deposited charge in each SV. However, even the MSV method has an issue to be addressed. Fig. 1 shows the charge collection efficiency at three locations in Fig. 7 as a function of the amount of deposited charge. The details of the TCAD model will be explained in Section 4. Fig. 1 reveals that the charge collection efficiency nonlinearly depends on the amount of charge. Therefore, it is difficult to construct an MSV model that can cover a wide range of charge deposition events. This issue is especially critical when considering that neutrons in the terrestrial environment and protons in space involve nuclear reactions producing various ions. Therefore, we conclude that the machine learning approach in [6, 7] offers promising potential.

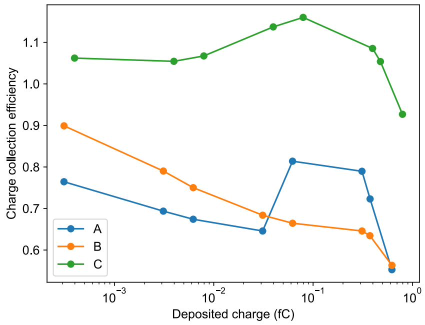

Figure 1: Relationship between deposited charge and charge collection efficiencies at points A–C in Fig. 7.

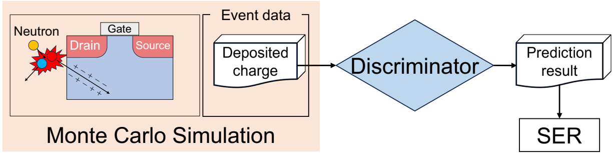

Figure 2: SER estimation flow.

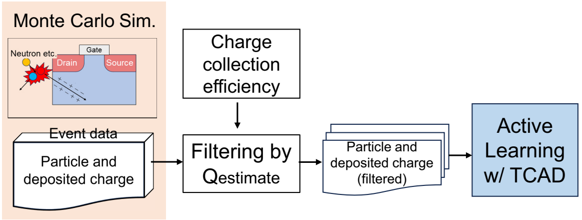

Figure 3: Proposed discriminator training with Monte Carlo simulation and active learning.

3 PROPOSED SEU DISCRIMINATOR CONSTRUCTION WITH ACTIVE LEARNING

3.1 Problem Definition

Fig. 2 shows the SER estimation method supposed in this work. The Monte Carlo particle transport simulation generates event data, each containing information on the charge deposited in devices. Each charge deposition event is then processed by the SEU discriminator to count SEUs and calculate the SER. More specifically, the charges deposited in multiple volumes within an SRAM cell serve as input features for the discriminator.

This work explores an efficient SEU discriminator construction method using TCAD simulations. Our goal in this paper is to maximize event-wise SEU inference accuracy without relying on empirical or experimental knowledge, while remaining constrained by the limited number of feasible TCAD simulations.

3.2 Proposed Method

3.2.1 Overview. To construct an SEU discriminator, we need to prepare a set of charge deposition data labeled based on SEU occurrence or non-occurrence. These labels are annotated using TCAD simulations. As discussed earlier, the number of TCAD simulations is limited, so careful selection of events for TCAD simulation is essential. This selection has two requirements: (1) the events should encompass diverse but plausible scenarios, and (2) the events should be mutually exclusive, maximizing the information available for training.

For the first requirement, we perform Monte Carlo particle transport simulations to generate a range of diverse events that can realistically occur. These Monte Carlo simulations are notably faster than TCAD simulations. For the second requirement, we use active learning to strategically select event data for additional labeling by TCAD simulation. However, the active learning-based selection initially does not perform well. To address this, we filter out events where an SEU is unlikely to occur. This filtering step is essential, as most particles, particularly neutrons and protons, do not cause SEUs. Without this filtering, the training data would be heavily biased, limiting the efficiency of early-stage training.

3.2.2 Detailed Procedure. Fig. 3 illustrates the proposed active learning-based training flow for the SEU discriminator. First, the Monte Carlo particle transport simulation is run to obtain a diverse dataset comprising incident particle information and the deposited charge in each SV. We assign multiple SVs [21] within a single transistor to enhance discrimination accuracy, with the method for determining these volumes detailed in Section 5.1. Generally, increasing the number of SVs allows for more precise modeling but requires larger training datasets. The particle transport simulator outputs information on incident particles, including particle type, energy, incident position, and direction vector, along with the charge deposited in each SV within the MSVs.

Secondly, we calculate Q_{estimate}, representing the collected charge estimated by summing the product of the charge deposited in each SV and its charge collection efficiency (see Section 5.1). We then filter out events where estimate Q_{estimate} is below a specified threshold. This threshold is set sufficiently lower than the critical charge to ensure an almost negligible likelihood of SEU occurrence.

Thirdly, the filtered data are given to the proposed active learning procedure. Fig. 4 shows the internal flow of active learning. We iteratively choose the event data, label it with TCAD simulation, and enrich the training dataset. We then train the discriminator using the available training dataset in each iteration. Next, all the unlabeled data are given to the current discriminator to compute the confidence of the discrimination. In the active learning scheme, the least confident event should be added to the training dataset since the accuracy of the discriminator is expected to improve near the least confident event data. Here, this is a binary classification task, so we define the confidence of prediction by how far $P(0, \text{non-upset})$ or $P(1, \text{upset})$ is from 0.5.

Another important consideration is updating hyperparameters in the machine learning algorithms. In the early stages, the available training data is limited, allowing only a small model to be constructed. However, as the training data grows, a larger model can be built, contributing to improved classification accuracy. Therefore, we periodically update the hyperparameters during the iterations.

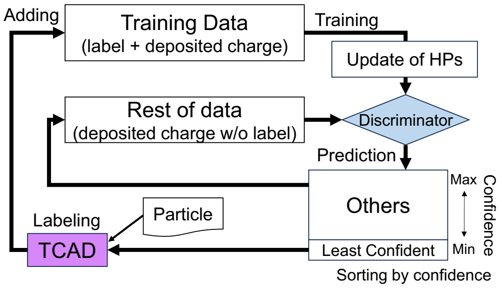

Figure 4: Internal flow of active learning with on-demand TCAD labeling and adaptive tuning of hyperparameters.

We also need to address the initial iteration, specifically the selection of event data for labeling. Through experimental investigation of various selection methods, we found that randomly selected data performs well, largely due to the previously mentioned filtering process. Therefore, in this work, we randomly choose the initial dozen events.

4 DUT AND SIMULATION SETUP

In this paper, we selected a 12-nm SRAM chip in a resin package mounted on a PCB as the device under test (DUT), where each SRAM cell consists of six FinFETs. We used PHITS [17] as the particle transport simulator and the Hyper Environment for Exploration of Semiconductor Simulation (HyENEXSS) [10] for the TCAD simulations.

The left figure of Fig. 5 shows the 3D structure of the SRAM cell used in the TCAD simulation. The gray areas represent silicon, while the white areas represent electrodes. The electrode arrangement corresponds to the layout of a high-density six-transistor SRAM bit cell [5]. Since cross-sectional views of 1x-nm FinFET transistors observed via transmission electron microscopy (TEM) are tapered [8], each fin is approximated by three stacked cuboids. Note that the simulation area is larger than an actual SRAM bit cell. The electrical characteristics of the SRAM cell were calibrated following the method described in [19]. A supply voltage V_{DD} is applied to the side of the N-well, and the substrate bottom is connected to GND. This model simulates bit-flipping behavior by performing transient analysis after electron-hole pairs are generated due to particle incidence.

Additional input information for each event data must be prepared for the TCAD simulation. As mentioned in Section 3, the dumped data contain information on particle type, energy, incident position, and direction vector. This information is used to deposit charge in the TCAD model. The amount of charge is proportional to the particle's linear energy transfer (LET), which depends on particle type and energy. We used The Stopping and Range of Ions in Matter (SRIM) code [23] for LET calculations.

Fig. 6 illustrates the chip-level DUT configuration used in PHITS, with SRAM transistors modeled according to Fig. 5. The SVs are positioned around the off-state NMOS, as it is the most sensitive transistor within the SRAM cell [14]. Detailed SV allocation will be discussed in Section 5.1. In this model, terrestrial neutrons irradiate

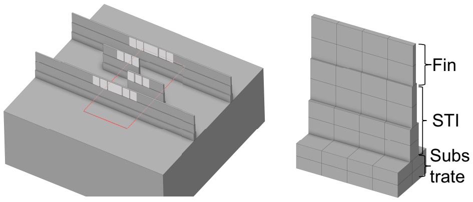

Figure 5: 3D TCAD simulation model of 12-nm FinFET 6T-SRAM cell. The red rectangle represents the cell boundary. The right figure explains sensitive volume allocation.

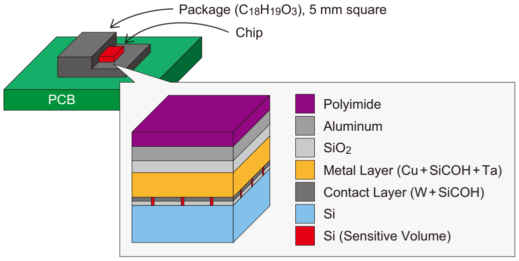
Package ($C_{18}H_{19}O_3$), 5 mm square
Chip
PCB
Polyimide
Aluminum
SiO_2
Metal Layer (Cu + SiCOH + Ta)
Contact Layer (W + SiCOH)
Si
Si (Sensitive Volume)

Figure 6: Constitution of the 12-nm board modeled in PHITS simulation.

the board from the package side, based on the ground-level neutron spectrum [15, 16]. The deposited charge data are recorded each time a particle enters the SVs.

5 EXPERIMENT

5.1 Multiple Sensitive Volume Allocation

When constructing the SEU discriminator, we allocate multiple sensitive volumes within an SRAM cell, with the charge deposited in each volume serving as an input feature. However, allocating volumes in insensitive regions is ineffective. To identify sensitive regions, we deposited a point charge of approximately 0.5 fC, comparable to the critical charge in similar generations [18], at various locations within the silicon of the off-state NMOS (including the substrate) and performed mixed-mode TCAD simulations. We then calculated the charge collection efficiency, defined as the charge collected by the drain divided by the deposited charge.

Fig. 7 shows the spatial distribution of charge collection efficiency at various points near the off-NMOS region. The right figure in Fig. 5 illustrates the entire SV, defined by a charge collection efficiency exceeding a threshold of 0.05. The first layer from the top is the fin, the second and third layers are silicon wrapped in shallow trench isolation (STI), and the fourth layer is part of the substrate. This volume was then divided into MSVs, as indicated by the lines in Fig.5. The first to third layers were each divided into eight volumes, and the fourth layer into 16 volumes, resulting in a total of 40 volumes. Using the charge collection efficiencies of these volumes, Q_{estimate} in Fig.3 is calculated. Note that this division is just an initial trial, with its optimization left for future work.

5.2 Data Preparation

We obtained the event data and filtered them by setting the Q_{estimate} threshold at 0.1 fC, which is several times smaller than the critical charge, as shown in Fig. 3. After filtering, the total number of events was 716. The proposed method annotates labels with TCAD simulation on the fly for active learning. However, applying this on-the-fly annotation to all comparison methods would lead to redundant TCAD simulations for the same events, resulting in inefficiencies. To prevent this, we preprocessed all event data with TCAD simulations and stored the results in advance. During training, label annotation is based on these pre-executed TCAD simulation results. Here, the TCAD simulations were performed with $V_{\mathrm{DD}} = 0.8$ V, which is the nominal voltage for this SRAM bit cell. Out of the total samples, 242 resulted in upsets, accounting for 33.8% of the data.

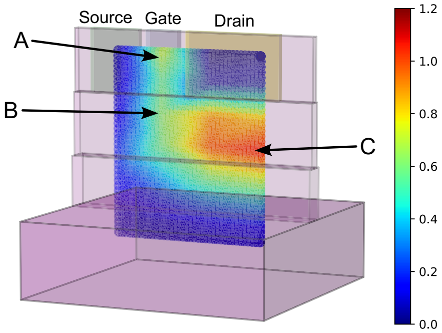

Figure 7: Charge collection efficiency around an off-state NMOS transistor obtained by TCAD simulation.

5.3 Discriminator Construction

We selected LightGBM [9] as the machine learning algorithm for the discriminator. The amounts of charge deposited in each SV shown in Fig. 5 are provided to LightGBM as features. LightGBM is a type of Gradient Boosting Decision Tree (GBDT) algorithm, which is widely used in machine learning tasks, including classification. We chose LightGBM because its training process is faster than that of other GBDT algorithms. Additionally, we tested other machine learning algorithms, such as random forests and neural networks, and found that LightGBM performed better in terms of accuracy, especially when the training data were limited. Results for the neural network model will be presented later.

During LightGBM training, we need separate training and test samples to prevent overfitting. We split the available data with a training-to-test ratio of 8:2, ensuring that the ratio of upset to non-upset samples is consistent across both sets.

In the first learning loop of Fig. 4, ten event dumps are randomly selected for label annotation and used as the initial training data. The active learning process is then repeated until all event data are labeled.

Additionally, for every ten new training samples added, the hyperparameters are updated according to the following procedure:

(i) Split the current training data into stratified 5-folds with randomness.
(ii) Explore hyperparameters using Optuna [4]. For each fold, construct a discriminator using the other four folds and evaluate its accuracy on the selected fold.
(iii) Repeat step (ii) 100 times and adopt the hyperparameters with the highest average accuracy across the 5-folds.

Table 1: LightGBM hyperparameters.

Parameter name	Lower limit	Upper limit	log
learning_rate	0.01	0.2	
lambda_l1	1e-8	10.0	True
lambda_l2	1e-8	10.0	True
num_leaves	2	256	
feature_fraction	0.01	1.0	
bagging_fraction	0.01	1.0	
bagging_freq	0	10	
min_child_samples	0	10	

Here, Optuna is an automatic hyperparameter optimization framework based on Bayesian optimization. The hyperparameters targeted for tuning and their ranges are listed in Table 1. The "log" column in the table indicates whether the parameter is scaled in the logarithmic domain. To obtain performance statistics, the discriminator is constructed 100 times using each construction method, with shuffling between training and test data.

5.4 Results

Fig. 8 shows the accuracy of the discriminator constructed by the proposed method as a function of the number of training samples, comparing it with various construction methods. The accuracy of the proposed discriminator is plotted as a solid blue line, with the blue-filled area representing its standard deviation. "AL" indicates that active learning is applied, while "Non-AL" indicates that it is not, meaning event data for labeling are selected randomly. Additionally, "HPupdate" indicates that hyperparameters are updated according to the above procedure, while "HPfix" indicates that hyperparameters are fixed, using the values obtained when steps (i)–(iii) are performed with all training data and kept constant throughout the iterative process.

As shown in the figure, both active learning and hyperparameter updating contribute to improved accuracy, especially when the amount of training data is small. Table 2 lists the number of training samples required to reach various accuracy levels. For example, the amount of training data needed to achieve 0.8 accuracy is reduced by 41% when both active learning and hyperparameter updating are enabled, compared to using neither. For 0.85 accuracy, the reduction is 31%. Additionally, it can be observed that hyperparameter updating contributes more to reducing the required training data than active learning.

For further comparison, we constructed the discriminator using neural networks with active learning and hyperparameter updating, represented by the green line in Fig. 8. The number of layers, the number of neurons per layer, the learning rate, the number of epochs, and the batch size were all tuned. The ReLU activation function was used in the hidden layers, while the output layer used a sigmoid activation. As shown, LightGBM provides more accurate predictions.

Furthermore, the accuracies of the conventional MSV and SSV methods are shown by the dotted red and purple lines, respectively. By sweeping the LET of an ion injected directly above the drain, the critical charges for MSV and SSV are set to the charge amount generated by the ion with the minimum LET that induces a bit flip in the TCAD simulation. The MSV method improves accuracy by 11% compared to SSV. Notably, the 68% accuracy achieved by the MSV method is reached by the proposed method with only 18 training samples.

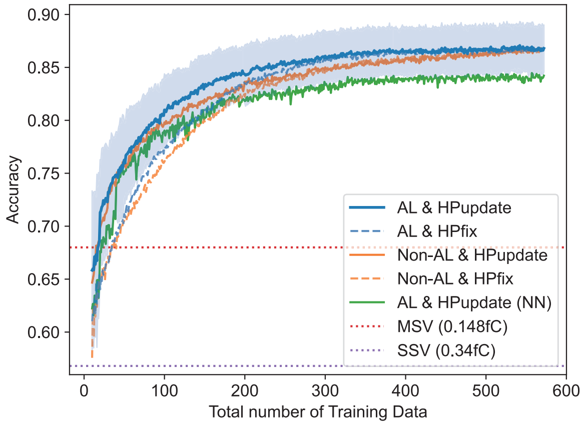

Figure 8: Accuracy of discriminators as a function of the number of training data. The area filled in blue represents the standard deviation for AL&HPupdate.

Table 2: Accuracy and required number of training data.

Accuracy	AL & HPupdate	AL & HPfix	Non-AL & HPupdate	Non-AL & HPfix
0.75	43	78	**41**	89
0.775	**59**	100	64	114
0.8	**88**	135	108	149
0.825	**133**	176	164	194
0.85	**215**	262	288	311

In addition, the F-score, defined as the harmonic mean of precision and recall, is also evaluated as an additional performance metric for the discriminator. Fig. 9 shows the F-scores of the discriminators. Similar to accuracy, both active learning and hyperparameter updating contribute to improvements in the F-score.

Finally, we discuss the runtime for constructing the discriminator. The average time required for a single TCAD simulation is approximately 12 hours on a Gen10 Xeon server. For 160 labeled event samples as training data, this translates to 1,920 hours. Even with 16 simulations running in parallel, it would still require five days. Therefore, the 30–40% reduction in training data achieved by the proposed method, as shown in the previous section, is highly significant. In contrast, the total time required to construct the proposed discriminator, including hyperparameter updates, is less than 30 minutes, and SER estimation takes only a few seconds. Thus, the runtime overhead from active learning and hyperparameter updating is well justified, given the substantial reduction in TCAD simulation time.

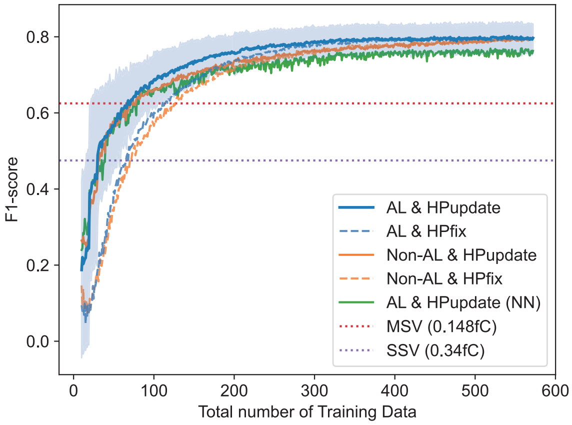

Figure 9: F-score of discriminators.

6 CONCLUSION AND FUTURE DIRECTION

In this paper, we proposed an SEU discriminator construction method featuring active learning and periodic hyperparameter tuning, designed to enhance both the accuracy and efficiency of SEU discrimination. A key advantage of the proposed method is that it systematically constructs the discriminator without requiring empirical determination of the SV and critical charge. We applied this method to a 12-nm FinFET SRAM using LightGBM. Experimental results demonstrate that both active learning and hyperparameter updating effectively reduce the amount of training data needed, achieving reductions of 41% and 31% when targeting 80% and 85% discrimination accuracy, respectively.

Let us discuss future work and directions. An urgent future task is to validate the proposed SER estimation using measurement data. We have irradiation data for neutrons, protons, and alpha particles, and reproducing these results is the next objective, with a particular focus on simulating MCU occurrences.

An important research direction is to extend the proposed method to account for particles striking multiple transistors within a cell. As noted in [7], such events alter sensitivity to charge deposition, and considering this factor is expected to improve event-wise SEU classification. In this context, GAA and CFET structures are likely to encounter these events more frequently due to their vertical proximity. Establishing SER estimation applicable to GAA and CFET structures is critically important for ensuring chip reliability.

Another approach to addressing the TCAD simulation time issue is to accelerate TCAD simulations. Recent work in [13] seeks to speed up TCAD simulations using machine learning. With sufficient acceleration, TCAD could be directly integrated into the Monte Carlo flow or used for generating training data in SEU discriminator construction, both of which are expected to enhance accuracy and improve estimation speed.

ACKNOWLEDGMENTS

This study is supported by JSPS KAKENHI Grant Number 24H00073.

REFERENCES

[1] Shin-ichiro Abe et al. 2012. Multi-Scale Monte Carlo Simulation of Soft Errors Using PHITS-HyENEXSS Code System. *IEEE Transactions on Nuclear Science* 59, 4 (2012), 965–970. https://doi.org/10.1109/TNS.2012.2187215

[2] Shin-Ichiro Abe et al. 2023. A Terrestrial SER Estimation Methodology Based on Simulation Coupled With One-Time Neutron Irradiation Testing. *IEEE Transactions on Nuclear Science* 70, 8 (2023), 1652–1657. https://doi.org/10.1109/TNS.2023.3280190

[3] S. Agostinelli et al. 2003. Geant4—a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506, 3 (2003), 250–303. https://doi.org/10.1016/S0168-9002(03)01368-8

[4] Takuya Akiba et al. 2019. Optuna: A Next-generation Hyperparameter Optimization Framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[5] David Burnett et al. 2014. FinFET SRAM design challenges. In *2014 IEEE International Conference on IC Design & Technology*. 1–4. https://doi.org/10.1109/ICICDT.2014.6838606

[6] Masanori Hashimoto et al. 2017. Soft error rate estimation with TCAD and machine learning. In *2017 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*. 129–132. https://doi.org/10.23919/SISPAD.2017.8085281

[7] Soichi Hirokawa et al. 2016. Multiple sensitive volume based soft error rate estimation with machine learning. In *2016 16th European Conference on Radiation and Its Effects on Components and Systems (RADECS)*. 1–4. https://doi.org/10.1109/RADECS.2016.8093181

[8] Dick James. 2016. Moore's law continues into the 1x-nm era. In *2016 27th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC)*. 324–329. https://doi.org/10.1109/ASMC.2016.7491159

[9] Guolin Ke et al. 2017. LightGBM: a highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 3149–3157.

[10] Norihiko Kotani. 1998. TCAD in Selete. In *Simulation of Semiconductor Processes and Devices 1998*, Kristin De Meyer and Serge Biesemans (Eds.). Springer Vienna, Vienna, 3–7.

[11] P. C. Murley and G. R. Srinivasan. 1996. Soft-error Monte Carlo modeling program, SEMM. *IBM Journal of Research and Development* 40, 1 (1996), 109–118. https://doi.org/10.1147/rd.401.0109

[12] Takashi Nakamura et al. 2008. *Terrestrial Neutron-Induced Soft Errors in Advanced Memory Devices*. WORLD SCIENTIFIC. https://doi.org/10.1142/6661 arXiv:https://worldscientific.com/doi/pdf/10.1142/6661

[13] Rodion Novkin, Simon Thomann, and Hussam Amrouch. 2023. ML-TCAD: Perspectives and Challenges on Accelerating Transistor Modeling using ML. In *2023 ACM/IEEE 5th Workshop on Machine Learning for CAD (MLCAD)*. 1–4. https://doi.org/10.1109/MLCAD58807.2023.10299886

[14] Philippe Roche et al. 1999. Determination of key parameters for SEU occurrence using 3-D full cell SRAM simulations. *IEEE Transactions on Nuclear Science* 46, 6 (1999), 1354–1362. https://doi.org/10.1109/23.819093

[15] Tatsuhiko Sato. 2015. Analytical Model for Estimating Terrestrial Cosmic Ray Fluxes Nearly Anytime and Anywhere in the World: Extension of PARMA/EXPACS. *PLOS ONE* 10, 12 (12 2015), 1–33. https://doi.org/10.1371/journal.pone.0144679

[16] Tatsuhiko Sato. 2016. Analytical Model for Estimating the Zenith Angle Dependence of Terrestrial Cosmic Ray Fluxes. *PLOS ONE* 11, 8 (08 2016), 1–22. https://doi.org/10.1371/journal.pone.0160390

[17] Tatsuhiko Sato et al. 2024. Recent improvements of the particle and heavy ion transport code system – PHITS version 3.33. *Journal of Nuclear Science and Technology* 61, 1 (2024), 127–135. https://doi.org/10.1080/00223131.2023.2275736

[18] Norbert Seifert et al. 2015. Soft Error Rate Improvements in 14-nm Technology Featuring Second-Generation 3D Tri-Gate Transistors. *IEEE Transactions on Nuclear Science* 62, 6 (2015), 2570–2577. https://doi.org/10.1109/TNS.2015.2495130

[19] Kozo Takeuchi et al. 2023. Voltage Dependence of Single-Event Cross Sections of FinFET SRAMs for Low LET Condition. *IEEE Transactions on Nuclear Science* 70, 8 (2023), 1755–1759. https://doi.org/10.1109/TNS.2023.3295340

[20] Kevin M. Warren et al. 2007. Application of RADSAFE to Model the Single Event Upset Response of a 0.25 μm CMOS SRAM. *IEEE Trans. Nuclear Science* 54, 4 (2007), 898–903. https://doi.org/10.1109/TNS.2006.889810

[21] Kevin M. Warren et al. 2007. Monte-Carlo Based On-Orbit Single Event Upset Rate Prediction for a Radiation Hardened by Design Latch. *IEEE Transactions on Nuclear Science* 54, 6 (2007), 2419–2425. https://doi.org/10.1109/TNS.2007.907678

[22] Robert A. Weller et al. 2010. Monte Carlo Simulation of Single Event Effects. *IEEE Transactions on Nuclear Science* 57, 4 (2010), 1726–1746. https://doi.org/10.1109/TNS.2010.2044807

[23] James F. Ziegler et al. 2010. SRIM - The stopping and range of ions in matter (2010). *Nuclear Instruments and Methods in Physics Research B* 268, 11-12 (June 2010), 1818–1823. https://doi.org/10.1016/j.nimb.2010.02.091