POLITECNICO DI TORINO
Repository ISTITUZIONALE

Vision Transformer Reliability Evaluation on the Coral Edge TPU

Original
Vision Transformer Reliability Evaluation on the Coral Edge TPU / Coelho, Bruno Loureiro; Bodmann, Pablo R.; Cavagnero, Niccolo; Frost, Christopher; Rech, Paolo. - In: IEEE TRANSACTIONS ON NUCLEAR SCIENCE. - ISSN 0018-9499. - 72:4(2025), pp. 1443-1451. [10.1109/tns.2024.3513774]

Availability:
This version is available at: 11583/2995241 since: 2024-12-12T13:00:07Z

Publisher:
IEEE

Published
DOI:10.1109/tns.2024.3513774

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

31 May 2026

Vision Transformer Reliability Evaluation on the Coral Edge TPU

Bruno Loureiro Coelho, Pablo Rafael Bodmann, Niccolò Cavagnero, *Member, IEEE*, Christopher Frost, and Paolo Rech, *Senior Member, IEEE*

***Abstract*— Vision transformers (ViTs) outperform convolutional neural networks (CNNs) in tasks such as image classification, and, despite their high computational complexity, they can still be mapped to low-power EdgeAI accelerators, such as the Coral tensor processing unit (TPU). In this article, through accelerated neutron beam experiments, we study the reliability of six ViTs on the Coral TPU and four microbenchmarks. According to our data, the internal size of attention heads (the main computational block in ViTs) has negligible impact on the failure-in-time (FIT) rate of the model compared to increasing the number of heads in the model; furthermore, our results show that employing convolutions in the patch embedding reduces the FIT rate of the model. Additionally, we decompose ViTs into four basic computational blocks that represent the main operators of the model, showing that, although the transformer layer [with multihead self-attention and multilayer perceptron (MLP)] presents the highest FIT rate, it is actually the patch embedding that is more likely to cause misclassifications. These results can be leveraged to design hardening techniques that improve the resilience of the critical blocks of a ViT, identified in our evaluation while minimizing the additional overhead.**

***Index Terms*— Failures-in-time (FIT), radiation, reliability, soft-errors, tensor processing unit (TPU), transformers.**

I. Introduction

PROCESSING visual information is a key task in applications such as self-driving cars, airplanes, space probes, and unmanned aerial vehicles (UAVs), where reliable computing is also crucial [1]. Until recently, convolutional neural networks (CNNs) were the main approach to detect or classify objects in an image or video. However, the accuracy of CNN-based detection is bounded by an intrinsic limitation due to the very nature of the convolution operation: being a local operator performed as a sliding window over the input image, the network can only extract information from pixels that are spatially close to each other. Attempts to increase the receptive field of CNNs [2], [3] have shown improvements in global reasoning capabilities at the expense of efficiency. Nonetheless, these approaches either introduce a significant information bottleneck [2] or enlarge the kernel size [3] without truly achieving a global receptive field. Fortunately, researchers have recently developed a new architecture capable of correlating input information on a global scale: the *transformer* model.

Transformers are a type of deep learning (DL) model architecture originally introduced in natural language processing (NLP), where they revolutionized the field. More recently, the transformer architecture has been successfully applied to image and video processing, being named vision transformers (ViTs). ViTs leverage the concept of *attention*, which allows global processing of information from all over the image, overcoming the spatially local receptive field of CNNs and resulting in higher accuracy. Interestingly, transformers, despite having a more complex architecture with respect to CNNs, can also be deployed in embedded applications with strict energy, weight, and space constraints. In this article, we study the reliability of transformer models on low-power and low-cost commercial-of-the-shelf (COTS) accelerators, such as the Coral Edge tensor processing unit (TPU), a device capable of processing neural networks (NNs) in an extremely cost-effective and energy-efficient manner.

While the effect of radiation on CNNs executed on TPUs has already been studied [4], [5], to the best of our knowledge, this is the first paper investigating the impact of atmospheric neutrons on the reliability of transformers running on TPUs. To provide a complete and accurate reliability overview, we consider six different ViT models: compact convolution transformer (CCT) [6], two standard ViTs [7] (one with eight attention heads and 8×8 patches, and another with 16 heads and 16×16 patches), and three EfficientFormers [8] (with increasing internal sizes, named L1, L3, and L7). Our data show that CCT has the lowest failure-in-time (FIT) rate, suggesting a reliability benefit in adopting convolution. Additionally, the FIT rate of the EfficientFormers does not depend on the model size, whereas ViT-16 has a $5\times$ higher FIT rate compared to the smaller ViT-8.

Additionally, to better understand the main reasons for the observed phenomena, we characterize the reliability of four microbenchmarks: two single-attention heads (one from ViT-8,

Received 3 October 2024; accepted 4 December 2024. Date of publication 9 December 2024; date of current version 18 April 2025. This work was supported in part by the Italian Ministry for University and Research (MUR) through the "Departments of Excellence 2023-27" Program awarded to the Department of Industrial Engineering under Grant L.232/2016 and in part by the European Union's 2020 Research and Innovation Program corresponding to the RADNEXT project under Grant 101008126. *(Corresponding author: Bruno Loureiro Coelho.)*

Bruno Loureiro Coelho and Paolo Rech are with the Department of Industrial Engineering, University of Trento, 38123 Trento, Italy (e-mail: bruno.loureirocoelho@unitn.it; paolo.rech@unitn.it).

Pablo Rafael Bodmann is with the Institute of Informatics, Universidade Federal do Rio Grande do Sul (UFRGS), Porto Alegre 91501-970, Brazil (e-mail: prbodmann@inf.ufrgs.br).

Niccolò Cavagnero is with the Department of Control and Computer Engineering, Politecnico di Torino, 10129 Turin, Italy (e-mail: niccolo.cavagnero@polito.it).

Christopher Frost is with ChipIR, Rutherford Appleton Laboratory, Science and Technology Facility Council, OX11 0QX Didcot, U.K. (e-mail: christopher.frost@stfc.ac.uk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNS.2024.3513774.

Digital Object Identifier 10.1109/TNS.2024.3513774

© 2024 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License.
For more information, see https://creativecommons.org/licenses/by/4.0/

and the other from ViT-16), and the transformer encoders from ViT-8 and ViT-16, respectively. As these microbenchmarks represent the most characteristic atomic operations of ViTs, they provide insights into how the architecture of each model affects the FIT rate. Furthermore, we evaluated eight additional micromodels that represent the main operations performed by the ViT model: patch embedding, multihead self-attention, transformer layer, and multilayer perceptron (MLP) classification head.

Overall, we present experimental data on 18 configurations of ViTs tested for more than 266 h of effective neutron irradiation at the ChipIR facility. When scaled to the natural neutron flux at New York City [9], this accounts for more than 258 billion years of neutron exposure. Our results show that the probability of radiation-induced errors affecting the output of a model increases with the model size. Furthermore, the probability of these errors is significantly affected by the complexity of the architecture, with more complex architectures such as the ones used in the EfficientFormers [8] being more susceptible to radiation effects. In addition to characterizing the reliability of different ViTs on the Coral EdgeTPU, we identify the most critical blocks of the transformer architecture. Specifically, our experimental data show that radiation-induced errors on the patch embedding layer of the transformer model are more likely to lead to misclassifications than errors on other layers of the model. Fortunately, our analysis has also shown that employing convolution operations on the patch embedding layer improves the resilience of the ViT model. These results can be used to design effective selective hardening techniques that improve the overall reliability of the model or to tune existing reliability solutions specifically designed for machine learning (ML) models [10].

The remainder of this article is structured as follows. Section II presents background information and related work. Next, we describe the experimental methodology in Section III. The results of our experiments are discussed in Section IV, where we characterize the reliability of transformer models and microbenchmarks. Finally, Section V concludes this article with our final remarks.

II. Background and Related Work

In this section, we present background information on the effects of radiation on NNs and discuss related work. Additionally, we provide details on the ViT architecture, and on the Coral Edge TPU.

A. Effects of Radiation on Neural Networks

Radiation-induced transient faults have three possible outcomes: 1) the fault propagates into an error that causes a *detected unrecoverable error* (DUE): a program crash or hang, thus requiring a restart of the application or the device; 2) the fault propagates through the stack of system layers and leads to silent data corruption (SDC), affecting the application output; or 3) the application is unaffected (i.e., the fault is masked, or the corrupted data is not used) [11]. The probability of radiation causing SDCs or DUEs depends on a combination of factors, including the *hardware architecture* (such as the memory/logic sensitivity [12], [13]) and the *application* [14]. As such, there is a need to study the reliability of a *given application* implemented on the *selected hardware* to safely deploy the system.

NNs are being applied to solve various tasks in several fields, such as computer vision and robotics. An NN is based on artificial neurons, which receive weighted inputs and apply an activation function to produce an output. While each neuron is relatively simple, a large number of neurons in parallel, called a *layer*, can process complex information. DL stacks several of these layers in sequence to build powerful models capable of achieving super-human performance in specific tasks [11].

While NNs can achieve high accuracy in image classification tasks, radiation-induced faults can negatively affect the model by causing SDCs. However, considering the output of an NN is probabilistic, the corrupted output can still allow for a correct classification. This can happen, for instance, when the corruption modifies classification probabilities without changing the class with the highest probability. Therefore, SDCs that do not affect the final classification are considered *tolerable* SDCs. In contrast, some SDCs *do* change the classification, thus being considered *critical* SDCs.

B. Vision Transformers

Transformers are the current state-of-the-art in ML models, being able to outperform previous architectures in multiple tasks across several fields, such as computer vision and robotics. ViTs [7] were shown to outperform CNNs, the previously most commonly adopted architecture for image processing. The improvement in accuracy achieved by ViTs is, in large part, due to its ability to process the entire image at once. In contrast, CNNs are intrinsically limited due to convolution being a local operation, thus binding the maximum achievable accuracy [7].

Fig. 1 illustrates a simplified architecture of the standard ViT [7], which adapts an architecture initially developed for NLP tasks to be able to process images. The basic ViT architecture splits the input picture into nonoverlapping patches, which are then encoded with information about their spatial position in the image. After this initial encoding, the data is processed by a series of *transformer layers*, responsible for the extraction of information from the input. Each transformer layer processes the input through a combination of *self-attention heads* and an MLP. Each attention head leverages the concept of self-attention to capture both global and local dependencies in the input data. More specifically, the self-attention mechanism weighs the importance of different patches in an image with respect to every other patch. This allows the model to identify and focus on the more complex and relevant relationships in the image. Therefore, the attention head is one of the core components of the ViT architecture, being responsible for identifying the main informative features of the input, thus affecting the final classification. After computing the attention scores, the transformer block leverages an MLP to increase the nonlinear fitting capability of the model. This process is repeated for each of the transformer layers in the model, with the output of one layer being used

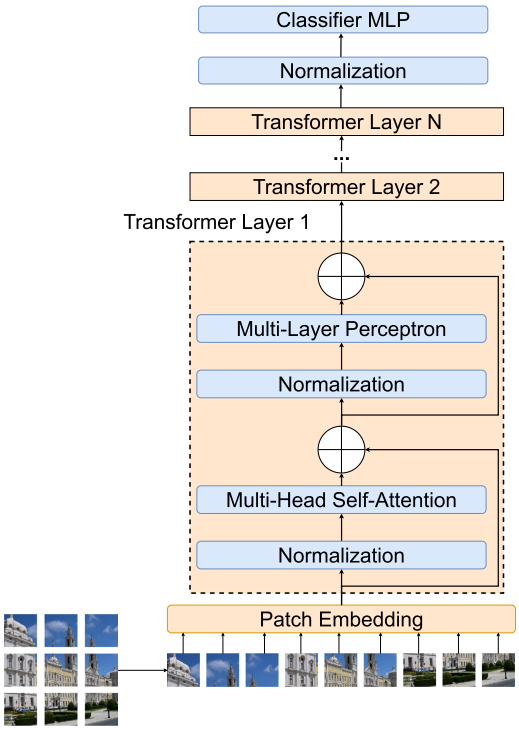

Fig. 1. Architecture of a ViT, adapted from [7].

as the input of the next layer. Finally, the output of the last transformer layer is forwarded to a classifier MLP responsible for outputting a prediction (class of an object). For a more in-depth understanding of the ViT architecture, we refer the reader to the original ViT paper [7].

Given the complex and heterogeneous architecture of ViTs, the effects of radiation may vary depending on the details of the architecture and model configuration. Therefore, considering the high accuracy and popularity of these models, we evaluate the reliability of six different **ViT models**, and four **microbenchmarks** that represent the core of the ViT architecture: single *attention heads* with two different internal sizes, and *transformer encoders* with two different configurations. Furthermore, to advance the understanding of the causes of *critical SDCs*, we evaluate **micromodels**, that is, partial models where we can observe intermediate outputs rather than only the final classification. The details of each model, microbenchmark, and micromodel are more thoroughly described in Section III-B.

As ViTs process the entire input at once, utilizing ViTs in real-time applications requires considerable computational power. Fortunately, ViTs can be mapped to low-power Edge AI accelerators that allow applications to achieve accurate image classification in a highly efficient manner. By using these devices, it is possible to perform visual tasks in embedded applications that have power constraints and high accuracy requirements, including safety-critical scenarios such as self-driving cars, airplanes, space probes, and nano-sats.

C. Coral Edge TPU

The Coral Edge TPU is a co-processor specialized in accelerating NNs, making it a promising candidate to be deployed in embedded applications, where power efficiency and performance are a requirement. While information and documentation about the technology of the TPU are sparse, we can say that this device is likely manufactured in 12- or 16-nm finFET technology, based on data published by Google [15], [16]. A Coral TPU is capable of computing 4 Tera operations per second, with a maximum consumption of 2 W, making it a highly efficient accelerator. To this end, the TPU operates over unsigned 8-bit integers, thus reducing the latency of data transfer between host and TPU, and additionally improving efficiency. As sensor data is usually in a floating point, the input data first goes through a dedicated quantization layer before being processed by the TPU. Once the accelerator finishes its computations, the output then goes through a de-quantization layer before being easily accessible by the host device. As the TPU only operates on unsigned 8-bit integers, the quantization layers are implemented by the host device, which is fortunately a negligible overhead due to the simplicity of the layers.

In our experiments, we adopt a Coral USB accelerator Edge TPU attached to a host Raspberry Pi 4. This setup has two main advantages, as the Raspberry Pi 4 represents a realistic embedded application scenario, while also allowing the neutron beam to target the TPU without irradiating the host device. Thus, this setup enables us to easily evaluate the reliability of the TPU without introducing errors in the host device, as the latter is not irradiated.

Fig. 2 shows an overview of the architecture of the Coral Edge TPU, which is composed of a systolic array fed by a large set of input buffers that do not have any kind of error protection. The systolic array applies the model's weights on the input of each layer before forwarding them to the activation unit. This unit accumulates the partial sums (of inputs multiplied by weights) and then applies the activation function, generating the output of the layer. The output of each layer is then used as the input of the next layer, which repeats the process with the respective weights and activation functions of each layer. After the final layer, the output is sent back to the host device, which applies the de-quantization layer and returns the floating-point output to the application.

To accelerate an NN with a TPU, the model must be first converted into an appropriate format. The framework for the Coral Edge TPU leverages TensorFlow and TensorFlowLite, a collection of libraries available in C++ and Python. The workflow necessary to map an NN into the TPU starts with a regular TensorFlow model, in 32-bit floating-point precision. Once the model is defined and trained, TensorFlowLite enables the conversion of the model to a quantized version that adopts an unsigned 8-bit integer. Finally, this quantized model is then converted into a TPU-compatible model by the Coral TPU compiler, which is provided by Google. Through quantization, ViTs severely reduce the computational cost of a TPU while maintaining an accuracy comparable with the original ViT. Once a model is successfully converted and compiled to run on the Coral TPU, utilizing the model is a straightforward process. First, the host application (e.g., Python script) instantiates a Coral interpreter, which then loads the already-compiled model. Next, the application utilizes the Coral interpreter API

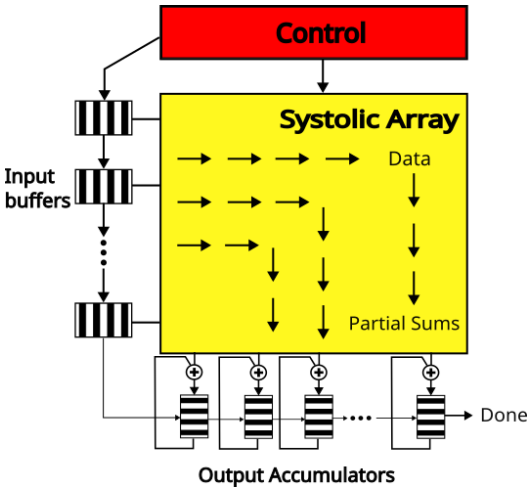

Fig. 2. Google's Coral Edge TPU architecture. Adapted from [17].

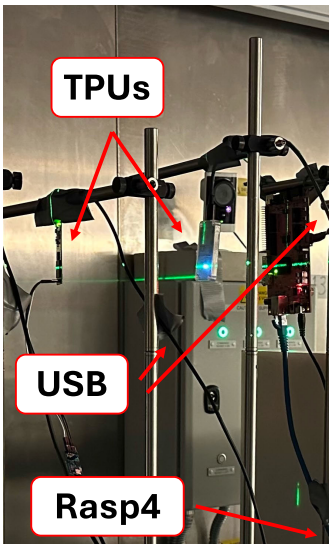

Fig. 3. Setup at ChipIR. The raspberry is out of the picture.

to load the input data into the TPU input buffers, which is achieved in a few lines of code. Finally, the host application requests the Coral interpreter to run inference on the TPU, an API call that returns when the outputs are already available via the interpreter API. At this point, the application can easily access the de-quantized output in floating point, thus making it readily usable.

D. Related Work

The reliability of the Coral Edge TPU has been studied with neutron [18], [19], [20], [21], heavy-ion [4], [15], proton [4], [22], and Co-60 [22] experiments with several applications, such as CNNs. Due to the TPU largely being a functional unit, as shown in existing research, radiation-induced errors often manifest as single-event upsets, thus leading to SDCs. Although less frequent, radiation can also cause recoverable or unrecoverable single-event functional interrupts, which, respectively, can or cannot run more applications without requiring a device reboot [15]. While these studies have shown promising results for the deployment of NNs on Edge TPUs, to the best of our knowledge, none of them have evaluated the reliability of *ViT models*. Due to the popularity of ViTs, some research has been carried out on their reliability [23], [24], [25], [26], but targeting other accelerators, such as GPUs. Hence, this work is the first paper to investigate the impact of atmospheric neutrons on the reliability of **ViTs running on TPUs**. As ViTs have significantly higher accuracy than CNNs [27], deploying ViTs on a Coral Edge TPU, an extremely power-efficient accelerator, presents opportunities for several applications, such as self-driving cars and space probes.

III. Methodology

In this section, we describe the experimental setup and the codes used for our evaluation.

A. Neutron Beam Experiments

The radiation experiments were performed at the ChipIR facility at the Rutherford Appleton Laboratory (RAL) in Didcot, U.K. ChipIR delivers a neutron beam suitable to mimic the atmospheric neutron effects in electronic devices [28], allowing the measurement of the FIT rate of the device executing a code. Fig. 3 shows part of our setup at ChipIR, where we irradiated four TPUs using a 3×3 cm beam spot, which is sufficient to irradiate the chip uniformly. The available neutron flux was about 3.5×10^6 n/(cm^2/s), allowing us to acquire data equivalent to 1844 years of natural exposure in only 60 s.

B. Tested Codes and Experimental Setup

To provide an in-depth evaluation of ViTs, we selected six transformer models: ViT-8 and ViT-16, which are TPU versions of the standard ViT [7] with different configurations; CCT, a transformer that applies convolutions during patch embedding; and three EfficientFormers with increasing complexity, named L1, L3, and L7. The input images were taken from the CIFAR-100 dataset and enlarged to 64×64 pixels during both training (preparing) and inference (evaluation) of the models. A list of each of the transformer models and their main characteristics follows.

1) ViT-8, the classical ViT model as described in [7], with eight heads with 128 channels and 8×8 patches.
2) ViT-16, a ViT with 16 heads with 256 channels and 16×16 patches.
3) CCT [6], a modification of the original ViT-8 (eight heads) that adopts convolutions to create and tokenize the image patches.
4) EfficientFormer [8], architectures that include a more advanced and efficient transformer block. We choose three models from this family with increasing internal sizes (L1, L3, and L7). All of them have eight attention heads. L3 is 2.54× larger than L1, whereas L7 is of slightly increased size (2.59× larger than L1).

ViT-8 and ViT-16 are chosen to understand the impact of the number of heads in the ViT sensitivity. CCT is tested to measure the effect of convolution in the error rate of a transformer, and the three EfficientFormers are chosen to

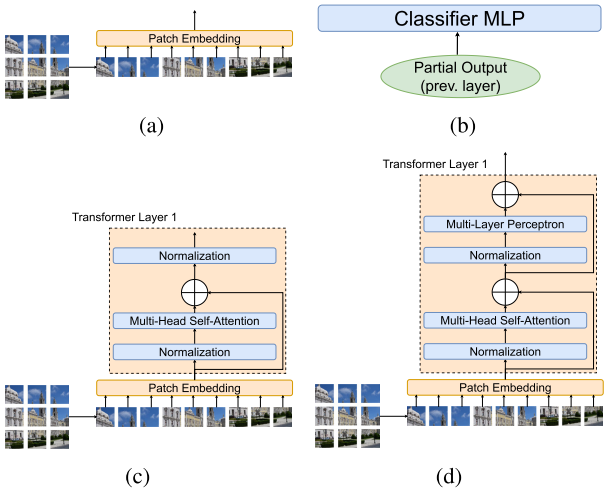

Fig. 4. Selected micromodels for our ablation experiments. Each micromodel allows us to gain insights into vital blocks of ViTs. (a) Patch embedding micromodel. (b) Classifier MLP micromodel. (c) Multihead self-attention micromodel. (d) Transformer layer micromodel.

compare a more efficient transformer block with increasing complexity.

Due to limitations on the TPU, the attention heads were implemented from scratch rather than using the ones available in TensorFlow. This was done because the Gaussian error linear unit (GELU) [29] activation function used by the MLPs is not supported by the TPU compiler. Thus, to overcome this limitation, we use the Tanh approximation [29], which can be mapped to the TPU without impacting the model accuracy.

Besides the transformer models, we also evaluated microbenchmarks, which are characteristic atomic operations executed in the ViT models. The microbenchmarks comprise two different single-attention heads with the same sizes as the ones in ViT-8 (Attention 1) and ViT-16 (Attention 2), and two transformer encoders, which also follow the sizes of the ones in ViT-8 and ViT-16 (listed as Transformer Encoders 1 and 2, respectively).

Additionally, to analyze how radiation affects different parts of the ViT model, we selected four *micromodels*, as shown in Fig. 4. The idea is to propose an ablation study, incrementally adding parts of the ViT model to understand the contribution of each part to the overall framework error rate. These micromodels were selected to evaluate several aspects of the ViT model: first, (a) *patch embedding* is responsible for creating an efficient representation of the image (which has a high dimensionality) while retaining the necessary information for the subsequent blocks of the model. Next, we wanted to isolate the (b) MLP classifier, the block responsible for outputting the classification scores. After evaluating the very first and last blocks, we selected the (c) *multihead self-attention* block, which computes the attention scores that indicate which parts of the image contain different kinds of relevant information. This information is then passed through an MLP to increase the nonlinear fitting capability of the model, which completes a single (d) *transformer layer*. In other words, the (d) transformer layer includes the (c) multihead self-attention block and an additional MLP (along with residual and normalization layers, as previously described and shown in Fig. 1).

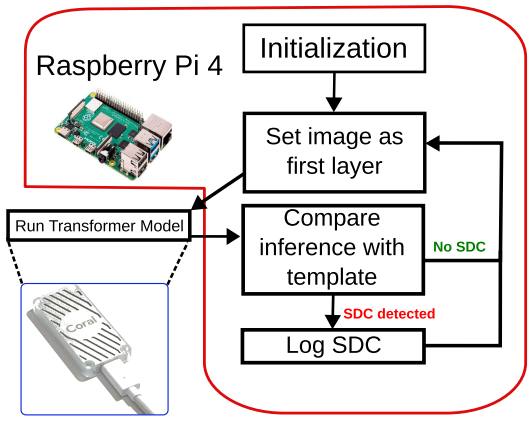

Fig. 5. Flow of an iteration of the experiment for the transformers.

The micromodels selected allow us to obtain and analyze the intermediate outputs of the model, which would be an otherwise impossible or highly inefficient process due to the way the Edge TPU functions. Particularly, the Edge TPU does not allow us to obtain intermediate results of an NN without executing part of the model on the host device. Therefore, obtaining intermediate results without micromodels requires synchronizing the TPU and host after every single layer to exchange the outputs and inputs of each layer (which also requires quantization of values). Instead, our approach allows us to obtain the output from each micromodel, which can then be used for further analysis.

The experiment consists of each TPU running one of the models, microbenchmarks, or micromodels listed above. Additionally, each host device (Raspberry Pi 4) only has one TPU connected via USB. Therefore, each host only runs one benchmark at a time. Fig. 5 shows an iteration of the experiment, which starts with the TPU being initialized with the model parameters, the test images, and the expected (*golden*) output for each image. After the initialization, the main loop starts: the image is fed as input to the TPU, which will then apply the model over that input. When the TPU completes its computations, it returns the output to the host device, which in turn compares the obtained result with the respective fault-free golden (expected) output. If there is any discrepancy between the computed output and the golden output, the erroneous data is logged for posterior analysis. After all the images of the batch have been tested, the main loop starts again from the first image. Considering that only the layers of the NN are executed on the TPU, whereas the comparison is executed on the Raspberry Pi (not irradiated), one can assume that all observed errors come from the TPU.

IV. Experimental Results

In this section, we present the results of neutron experiments with several transformer models and microbenchmarks. Section IV-A shows how radiation-induced errors affect transformers on the Edge TPU, while Section IV-B analyzes how errors in different layers affect the correct application output.

A. Radiation-Induced Errors on Edge TPU Transformers

To better understand how radiation affects ViTs running on Edge TPUs, we evaluated both entire transformer

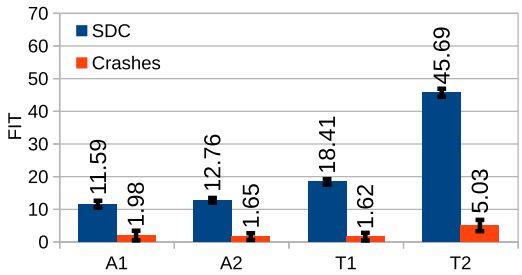

Fig. 6. FIT rate for the microbenchmarks. A1 and A2 are single-attention heads of size 128 and 256, respectively. T1 and T2 are transformer encoders of eight heads of size 128, and 16 heads of size 256, respectively.

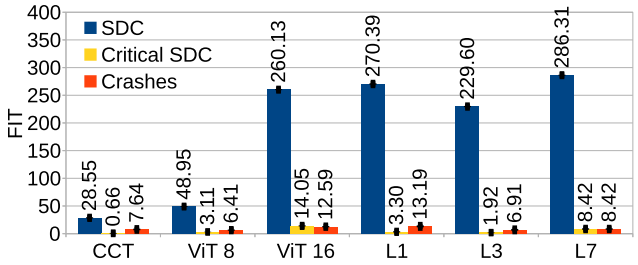

Fig. 7. FIT rate for the tested models. CCT uses convolution, ViT 8 and 16 are the classical transformers, and L1, L3, and L7 are EfficientFormers with increasing complexity.

models and microbenchmarks that compose the core computations of transformers. While evaluating entire models allows us to characterize the reliability of different configurations and architectures, evaluating microbenchmarks provides more detailed information about which parts of a model are more susceptible to radiation-induced faults.

Fig. 6 shows the SDC (blue) and DUE (red) FIT rates measured for each of the four microbenchmarks previously described in Section III-B. The data are plotted with 95% confidence intervals considering a Poisson distribution (we collected at least 100 events per microbenchmark). As expected, the FIT rate for DUEs is significantly lower than the FIT rate for SDCs, which is a characteristic of the Edge TPU: the separation between the accelerator (TPU) and the host (Raspberry Pi 4) makes DUEs less likely than in embedded systems. Additionally, the drivers of the Coral Edge TPU run entirely on the host device (Raspberry Pi 4), further increasing the robustness of the accelerator to DUEs. As the DUE FIT rate is both low and consistent in every model evaluated, the rest of our analysis will focus on radiation-induced SDCs.

As previously described, *Attention 1* and *Attention 2* are single self-attention heads, where each self-attention head identifies relationships between image patches on both local and global scales. The difference between the two self-attention heads is the internal size used: Attention 1 matches the size of the heads used in ViT-8 (internal size of 128), and Attention 2 matches the size of ViT-16 (internal size of 256). Similarly, *Transformer Encoder 1* and *Transformer Encoder 2* differ by the number of heads and internal size used: the first matches those of ViT-8 (eight heads of size 128), whereas the second matches those of ViT-16 (16 heads of size 256). The transformer encoder is the main block of the model, combining the information from each self-attention head and MLP.

Based on the results shown in Fig. 6, despite the different sizes, the FIT rates of Attention 1 and Attention 2 are similar. However, this is not true for the transformer encoders: Transformer Encoder 2 has a 2.51× higher FIT rate than Transformer Encoder 1. As the comparison between attention heads showed that the internal size has a negligible effect on the FIT rate, we can deduce that the difference between the transformer encoders is due to the increased number of heads (eight heads in Transformer Encoder 1 and 16 in Transformer Encoder 2). Therefore, transformers with a smaller number of heads are likely to be more reliable due to the lower probability of radiation-induced errors. However, it is important to note that reducing the number and size of heads may decrease the accuracy of the model: while ViT-16 (16 heads of size 256) has an accuracy of 97%, ViT-8 (eight heads of size 128) achieves only 93% accuracy.

In addition to the microbenchmarks, we also evaluated the reliability of six ViT models to investigate how radiation affects each architecture. Fig. 7 shows the FIT rates measured for the models previously explained in Section III-B: CCT uses convolutions in conjunction with self-attention heads; ViT-8 and ViT-16 are the baseline ViT model with different internal configurations (ViT-16 is larger); and *EfficientFormer L1, L3, L7* each use eight heads of increasing complexity (i.e., EfficientFormer L7 has more complex heads than L3 and L1).

While both CCT and ViT-8 have eight attention heads, ViT-8 has a 1.71× higher SDC FIT rate. Since the internal size of each head has little impact on the FIT rate (as previously shown in Fig. 6), this difference can be attributed to the way the input image is processed: CCT uses convolutions to create and tokenize the patches of the input image, whereas ViT-8 directly splits the image and creates tokens from image patches. Since convolutions are applied in an overlapping manner over neighboring pixels, each patch includes some information from adjacent patches. This redundancy of information could explain why convolutions improve the resilience of the model, as even if a patch is corrupted, part of its information is encoded in neighboring patches. Convolutions, then, not only help in improving the training efficiency but also reduce the model SDC FIT rate.

In line with the evaluation of microbenchmarks, the results for the full models, shown in Fig. 7, confirm that larger models are more sensitive to neutrons: ViT-16 has an FIT rate 5.31× higher than ViT-8. This difference is mainly due to the higher number of heads in ViT-16, as seen in the comparison between the Transformer Encoder 1 and Transformer Encoder 2 microbenchmarks. Additionally, the SDC FIT rates of the three EfficientFormers are very similar to each other, which can be attributed to the models having the same number of heads. While they have different internal sizes, the results for the EfficientFormer models agree with the evaluation of microbenchmarks, which show that the size of the attention head has negligible impact on the SDC FIT rate.

Surprisingly, the three EfficientFormers have a very similar SDC FIT rate to ViT-16 despite the latter having double the number of heads. This result can be explained by architectural differences between ViTs and EfficientFormers, as the

latter uses several inverted residual blocks (similar to the ones inside the MobileNet [30] CNN), including a modified block with the attention heads. Additionally, EfficientFormers run patch embedding multiple times—once at the start of each "MetaBlock" (the main blocks of the EfficientFormer architecture). In future work, we plan to evaluate the reliability of the different blocks of EfficientFormers (microbenchmarks).

In addition to the SDC FIT rate (blue) shown in Fig. 7, we also measured the *critical* SDC FIT rate (yellow). A critical SDC is defined as a wrong classification, that is, the class detected is not the expected one. In contrast, a *tolerable* SDC is a change in classification probabilities without changing the predicted class, meaning that any number of prediction probabilities differed from what was expected, yet the final classification was not changed. Our evaluation shows that the critical SDC FIT rate (yellow) is quite low, meaning that neutrons have a low chance of modifying the final classification. Further analysis of critical SDCs (not shown in Fig. 7) revealed that the expected class is still among the classes with the five highest probabilities, meaning that transformer models are capable of extracting some correct information even through radiation-induced errors.

We further analyzed tolerable (noncritical) SDCs by measuring how much the probability of the detected class has changed. The results showed that, in tolerable SDCs, ViTs present considerable differences in the probability of the expected class: 4.08% in ViT-8 and 4% in ViT-16. In contrast, the effect on CCT and the EfficientFormers is lower: 1.4% in CCT, 1.8% in EfficientFormer L1, 1.5% in L3, and 2.39% in L7. These results indicate that CCT is the most robust network among the ones tested, as it presented the lowest FIT rate, lowest critical SDC FIT rate, and little change in the probability of the expected class on the noncritical SDCs.

B. Identifying the Causes of Misclassifications

On top of characterizing the reliability of microbenchmarks and models (in the discussion above), we aimed to identify the causes of misclassifications (critical SDCs). To this end, we carefully selected and evaluated select parts (or *micromodels*) of the ViT model, as previously detailed in Section III-B: the *patch embedding*, the first *multihead self-attention* (including residual and normalization layers), the first *transformer layer*, and the *final MLP classifier*. By evaluating these blocks, we can identify the parts of the model that are more vulnerable to radiation, both in terms of the probability of an error happening (SDC FIT rate), and how an error propagates to the final output (how likely it is to cause a critical SDC). Our experiments included each of the four blocks for the ViT-8 and ViT-16 configurations, except for the patch size: every micromodel used a patch size of 8×8. By using a constant patch size, we were able to further analyze the impact of the size of the internal dimensions of the transformers.

First, we analyzed the average error magnitude for tolerable and critical SDCs to determine if there is a significant difference depending on the criticality of the SDC. Fig. 8 shows the average error magnitude for the ViT-8 and ViT-16 (full) models, with a higher value meaning that the incorrect output is further from the correct (expected) value. The data show

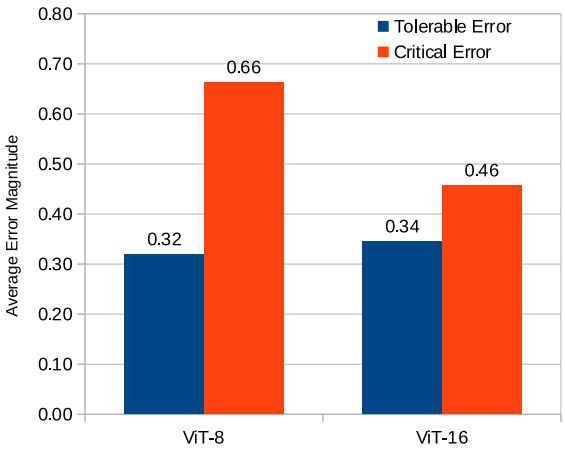

Fig. 8. Average error magnitude (relative to correct value) for ViT-8 and ViT-16 models.

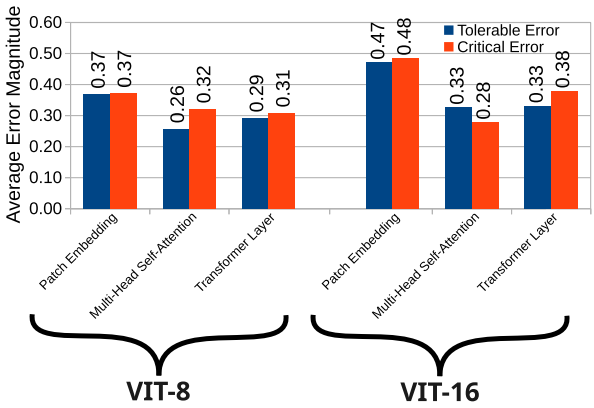

Fig. 9. Average error magnitude (relative to correct value) for ViT-8 and ViT-16 blocks (microbenchmarks).

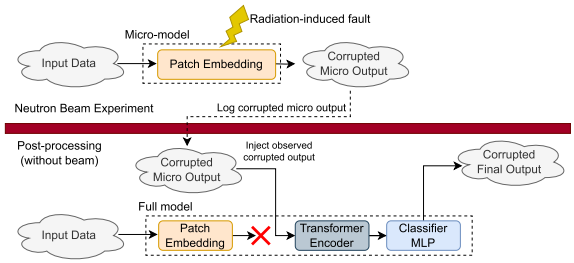

Fig. 10. We utilize the corrupted outputs observed and collected during beam experiments to inject (real) errors in the full ViT models.

that critical SDCs have higher error magnitude than tolerable SDCs for both ViT-8 and ViT-16. This result can be intuitively explained, as larger differences in the class probabilities will likely lead to misclassification. In addition to the transformer models, we also analyzed the output of the micromodels, shown in Fig. 9. Surprisingly, the error magnitude for the output of micromodels does not vary significantly whether it is a tolerable or critical SDC. This could mean that the *initial value* value of the error does not have a large impact on whether it will cause a misclassification or not, but rather that the misclassification happens due to *how the error propagates*.

Next, we aimed to identify how errors in different parts of the transformer model propagate into the output. As previously explained in Section III-B, we cannot extract the output of every single layer during one execution on the Edge TPU.

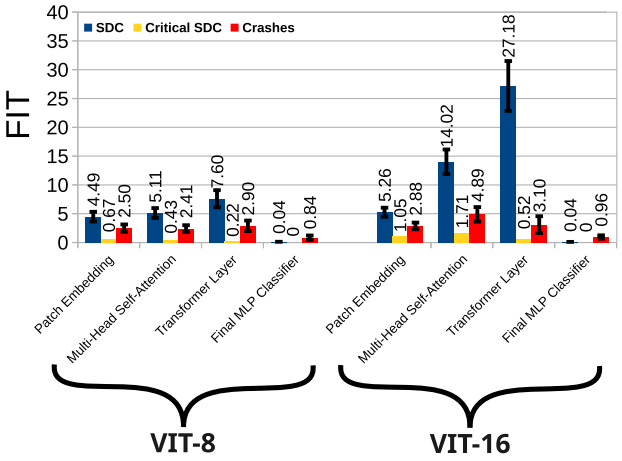

Fig. 11. SDC (tolerable and critical) and DUE FIT rates of the main parts of the ViT-8 and ViT-16 transformer models.

Therefore, as illustrated in Fig. 10, we leverage corrupted outputs of micromodels (gathered with real beam experiments) and inject them into the full model to obtain what would be the final corrupted output (i.e., the classification probabilities). While Fig. 10 shows an example of the patch embedding micromodel, the process is analogous for the other three micromodels, with the corrupted outputs collected during the experiment being injected into the appropriate part of the full model. This process allows us to observe intermediate errors while also being able to realistically simulate how the error would have affected the final output.

Fig. 11 combines the SDC FIT rate of each evaluated micromodel with the ratio of critical SDCs. Based on these results, while the transformer layer is the most likely to suffer radiation-induced faults, these SDCs rarely affect the final classification, that is, they are tolerable SDCs (meaning they are not critical). In contrast, errors in the patch embedding often lead to misclassifications (critical SDCs): around 15% of all SDCs in patch embedding lead to misclassification in ViT-8 and 20% in ViT-16. These results show that errors while processing the initial image are more critical than errors affecting the computation in the internal parts of the transformer encoder. Because the patch embedding reduces the dimension of the initial image with a large number of pixels (over 12 000 values for a 64×64 image) to a small number of patches (e.g., 192 patches for ViT-8), errors in this procedure may significantly affect the representation of patches. Additionally, errors in the first layer of the model will cascade into every subsequent layer, which is further aggravated by the nature of the patch embedding layer. Finally, patch embedding is the only nonresidual layer, meaning that radiation-induced errors are not smoothed out by re-using previously (correctly) computed data.

C. Impact of Experimental Results

Based on the results shown above, we can identify the patch embedding as the most critical operation in the ViT model, meaning that a radiation-induced error in this block has a high probability of resulting in a misclassification (critical SDC). Additionally, while transformer layers have the highest FIT rate of the evaluated blocks, most of these SDCs are tolerable, that is, they do not change the final classification. This is an important result considering the transformer encoder comprises the vast majority of the computations performed by ViT, as this block includes several transformer layers (three in the evaluated models). In contrast, the patch embedding requires orders of magnitude fewer operations and is only executed once in the ViT model. Thus, it may be possible to improve the reliability of ViT by implementing selective hardening techniques on this block. Due to it being a lightweight operation, replicating this block would introduce negligible overhead while protecting the most critical operation of the model. Alternatively, as shown in the comparison of ViT architectures, employing convolutions in the patch embedding increases the reliability of the model.

V. Conclusion

In this work, we reported the results collected after irradiating Coral Edge TPUs with neutrons for over 266 effective hours. We considered six different transformer models, four microbenchmarks, and eight parts of the ViT model (micromodels), for a total of 18 configurations evaluated. Data showed that the size of the head has a negligible influence on the model FIT rate, while the number of heads impacts the SDC FIT rate significantly. When comparing different models, the results indicate that the underlying architecture of the transformer has a large influence on the SDC FIT rate. By evaluating both microbenchmarks and full models, our experimental data provided valuable insights into the sensitivity of each part of ViTs to radiation-induced faults. Additionally, after observing real SDCs in different parts of the ViT model, we injected the errors collected during the experiment to determine what parts of the model are more likely to cause misclassifications. Based on this analysis, we identified the patch embedding as the most critical component of ViT. Interestingly, our experimental results have also shown that employing convolutions during patch embedding considerably improves the reliability of the model.

Acknowledgment

Neutron beam time was provided by ChipIR [DOI: 10.5286/ISIS.E.RB2400013] thanks to C. Cazzaniga, and M. Kastriotou.

References

[1] *Road Vehicles—Functional Safety*, ISO Standard 26262, Dec. 2018.
[2] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 7132–7141.
[3] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 11966–11976.
[4] M. C. Casey et al., "Single-event effects on commercial-off—The shelf edge-processing artificial intelligence ASICs," *IEEE Trans. Nucl. Sci.*, vol. 70, no. 8, pp. 1716–1723, Aug. 2023.
[5] R. L. R. Junior et al., "High energy and thermal neutron sensitivity of Google tensor processing units," *IEEE Trans. Nucl. Sci.*, vol. 69, no. 3, pp. 567–575, Mar. 2022.
[6] A. Hassani, S. Walton, N. Shah, A. Abuduweili, J. Li, and H. Shi, "Escaping the big data paradigm with compact transformers," 2021, *arXiv:2104.05704*.

[7] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent.*, Vienna, Austria, May 2021, pp. 1–22.

[8] Y. Li et al., "EfficientFormer: Vision transformers at MobileNet speed," in *Proc. Adv. Neural Inf. Process.*, vol. 35, New Orleans, LA, USA, 2022, pp. 12934–12949.

[9] C. Slayman, "JEDEC standards on measurement and reporting of alpha particle and terrestrial cosmic ray induced soft errors," in *Soft Errors in Modern Electronic Systems*. Boston, MA, USA: Springer, 2011, vol. 41, ch. 3, pp. 55–76.

[10] N. Cavagnero, F. D. Santos, M. Ciccone, G. Averta, T. Tommasi, and P. Rech, "Transient-fault-aware design and training to enhance DNNs reliability with zero-overhead," in *Proc. IEEE 28th Int. Symp. On-Line Test. Robust Syst. Design (IOLTS)*, Torino, Italy, Sep. 2022, pp. 181–187.

[11] P. Rech, "Artificial neural networks for space and safety-critical applications: Reliability issues and potential solutions," *IEEE Trans. Nucl. Sci.*, vol. 71, no. 4, pp. 377–404, Apr. 2024.

[12] R. C. Baumann, "Radiation-induced soft errors in advanced semiconductor technologies," *IEEE Trans. Device Mater. Rel.*, vol. 5, no. 3, pp. 305–316, Sep. 2005.

[13] J. Noh et al., "Study of neutron soft error rate (SER) sensitivity: Investigation of upset mechanisms by comparative simulation of FinFET and planar MOSFET SRAMs," *IEEE Trans. Nucl. Sci.*, vol. 62, no. 4, pp. 1642–1649, Aug. 2015.

[14] V. Sridharan and D. R. Kaeli, "Using hardware vulnerability factors to enhance AVF analysis," in *Proc. 37th Annu. Int. Symp. Comput. Archit.*, New York, NY, USA, Jun. 2010, pp. 461–472.

[15] M. Casey, E. Wyrwas, and R. Austin, "Recent radiation test results on cots AI edge processing ASICs," in *Proc. NEPP Electron. Technol. Workshop (ETW)*, Greenbelt, MD, USA, Jun. 2022, pp. 1–26.

[16] N. P. Jouppi et al., "Ten lessons from three generations shaped Google's TPUv4i: Industrial product," in *Proc. ACM/IEEE 48th Annu. Int. Symp. Comput. Archit. (ISCA)*, Jun. 2021, pp. 1–14.

[17] Q-Engineering. *Google Coral Edge Tpu Explained in Depth*. Accessed: Jan. 2, 2023. [Online]. Available: https://qengineering.eu/google-corals-tpu-explained.html

[18] R. L. R. Junior and P. Rech, "Reliability of Google's tensor processing units for convolutional neural networks," in *Proc. 52nd Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw.-Supplemental Volume (DSN-S)*, Baltimore, MD, USA, Jun. 2022, pp. 25–27.

[19] R. L. Rech and P. Rech, "Reliability of Google's tensor processing units for embedded applications," in *Proc. Design, Automat. Test Eur. Conf. Exhib. (DATE)*, Antwerp, Belgium, Mar. 2022, pp. 376–381.

[20] P. R. Bodmann and P. Rech, "Tensor processing unit reliability dependence on temperature and radiation source," *IEEE Trans. Nucl. Sci.*, vol. 71, no. 4, pp. 854–860, Apr. 2024.

[21] P. R. Bodmann, M. Saveriano, A. Kritikakou, and P. Rech, "Neutrons sensitivity of deep reinforcement learning policies on EdgeAI accelerators," *IEEE Trans. Nucl. Sci.*, vol. 71, no. 8, pp. 1480–1486, Aug. 2024.

[22] G. Lentaris et al., "Performance and radiation testing of the coral TPU co-processor for AI onboard satellites," in *Proc. Eur. Data Handling Data Process. Conf. (EDHPC)*, Juan Les Pins, France, Oct. 2023, pp. 1–4.

[23] K. Ma, C. Amarnath, and A. Chatterjee, "Error resilient transformers: A novel soft error vulnerability guided approach to error checking and suppression," in *Proc. IEEE Eur. Test Symp. (ETS)*, Venezia, Italy, May 2023, pp. 1–6.

[24] X. Xue et al., "Soft error reliability analysis of vision transformers," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 31, no. 12, pp. 2126–2136, Dec. 2023.

[25] L. Roquet, F. Fernandes dos Santos, P. Rech, M. Traiola, O. Sentieys, and A. Kritikakou, "Cross-layer reliability evaluation and efficient hardening of large vision transformers models," in *Proc. 61st ACM/IEEE Design Autom. Conf.*, San Francisco, CA, USA, Jun. 2024, pp. 1–6.

[26] G. Gavarini, A. Ruospo, and E. Sanchez, "Evaluation and mitigation of faults affecting Swin transformers," in *Proc. IEEE 29th Int. Symp. On-Line Test. Robust Syst. Design (IOLTS)*, Crete, Greece, Jul. 2023, pp. 168–174.

[27] L. Yuan et al., "Tokens-to-token ViT: Training vision transformers from scratch on ImageNet," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Guangzhou, China, Oct. 2021, pp. 538–547.

[28] C. Cazzaniga and C. D. Frost, "Progress of the scientific commissioning of a fast neutron beamline for chip irradiation," *J. Phys.*, vol. 1021, pp. 12037–12041, May 2018.

[29] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," Jun. 2023, *arXiv:1606.08415*.

[30] A. G. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

Open Access funding provided by 'Università degli Studi di Trento' within the CRUI CARE Agreement