# Tensor Operator Set Architecture (TOSA) in Machine Learning on ARM Platforms

The increasing demand for efficient and portable machine learning solutions, particularly in edge and embedded computing, has led to a complex landscape of diverse hardware platforms and software frameworks. Each framework, such as TensorFlow, PyTorch, and ONNX, offers its own set of operators for building neural networks 1. This heterogeneity creates significant challenges for developers aiming to deploy their models across a wide range of devices, especially the power-efficient and versatile ARM-based processors that are prevalent in these domains 1. The sheer number of operators, which continues to grow with the rapid evolution of machine learning, poses a substantial burden on hardware and software implementations, hindering device portability and efficient deployment 1.

The Tensor Operator Set Architecture (TOSA) has emerged as a pivotal solution to address these challenges. It provides a minimal yet stable set of tensor-level operators 1 that act as a common intermediate representation to which operators from various machine learning frameworks can be reduced 1. The primary goal of TOSA is to enable a variety of implementations that can run seamlessly on a diverse range of processors, including ARM's SIMD CPUs, Mali GPUs, and specialized neural processing units (NPUs) like the Ethos-U family, while ensuring consistent results at the TOSA level 3.

At its core, TOSA defines a comprehensive set of **whole-tensor operations** 3, B\_B1] that are fundamental to deep neural networks. These operations are not further decomposed within the TOSA specification, allowing for flexibility in how they are implemented on different hardware 1. The TOSA operator set is meticulously categorized to cover a broad spectrum of functionalities essential for modern neural networks. These categories include tensor operators for core computations like convolution (CONV2D), matrix multiplication (MATMUL), and pooling (AVG\_POOL2D, MAX\_POOL2D); activation functions such as SIGMOID and TANH; elementwise binary operators like ADD and MUL, as well as unary operators like ABS and NEGATE; reduction operators such as REDUCE\_SUM and REDUCE\_MAX; operators for manipulating data layout like RESHAPE and TRANSPOSE; scatter/gather operations (GATHER, SCATTER); image-specific operators like RESIZE; type conversion operators (CAST, RESCALE); data nodes for representing constants (CONST) and identity operations (IDENTITY); custom operators for extensibility (CUSTOM); control flow operators for conditional execution (COND\_IF) and loops (WHILE\_LOOP); variable operators for handling stateful computations (VARIABLE, VARIABLE\_WRITE, VARIABLE\_READ); and shape operators (CONST\_SHAPE).

Crucially, TOSA offers **full support for both quantized integer and floating-point content** 3, B\_B12. This dual support is vital for the ARM ecosystem, which spans from high-performance server processors to deeply embedded microcontrollers. Quantized models, in particular, are often preferred for deployment on resource-constrained ARM devices due to their reduced memory footprint and faster inference speeds 2. TOSA incorporates precision details directly into its operator definitions, effectively combining quantization information with the functional behavior of each operation 5.

Every operator within the TOSA specification is accompanied by a **precise functional and numerical description** 1. These descriptions meticulously detail the expected behavior of each operator, including the handling of numerical aspects such as precision, saturation, and scaling, especially for quantized datatypes 3. This level of detail ensures that developers can rely on consistent and predictable behavior across different hardware implementations, which is paramount for achieving model accuracy and reliability on diverse ARM platforms.

A fundamental design principle of TOSA is its **agnosticism towards any single high-level framework, compiler backend stack, or particular target hardware** 1. This independence allows machine learning frameworks to lower their respective operators into the standardized TOSA representation. Once in TOSA format, models can be processed by various compiler backends and ultimately executed on a wide range of ARM processors and accelerators without being tied to the specifics of the original framework or the final hardware target.

The TOSA specification itself undergoes a structured development process, employing a versioning scheme similar to software releases, with major, minor, and patch versions 3. This approach ensures a balance between stability and evolution, where major releases might introduce breaking changes, minor releases typically add new features in a backward-compatible manner, and patch releases focus on addressing bugs and providing clarifications 3.

The adoption of TOSA within the ARM ecosystem yields several significant benefits. **Portability** is greatly enhanced as applications and frameworks targeting TOSA can be deployed across a vast array of ARM processors, from the energy-efficient Cortex-M series to the high-performance Cortex-A series, as well as specialized NPUs like the Ethos-U family 3. This broad portability reduces the development effort required to target different ARM devices. **Efficiency** is another key advantage, as TOSA's focus on essential tensor operations and its inherent support for quantization enable the development of highly optimized implementations on ARM hardware 5. This leads to improved performance, lower latency, and reduced power consumption, which are particularly crucial for edge and embedded AI applications. Furthermore, TOSA offers a degree of **future-proofing** 1. As machine learning continues to evolve with new models and operators, if these can be effectively represented using the existing TOSA primitives, they can potentially run on TOSA-compliant ARM hardware already in deployment. Finally, TOSA's status as an **open standard** 2 with a permissive license 4 fosters community collaboration and innovation, benefiting the entire ARM machine learning ecosystem.

### Integration with Deep Learning Frameworks on ARM

The utility of TOSA is significantly amplified through its integration with popular deep learning frameworks when targeting ARM platforms.

**PyTorch Integration:** Arm has actively collaborated with Meta to provide strong support for ARM within ExecuTorch 8, an end-to-end solution designed to streamline the deployment of PyTorch-developed AI models on ARM-powered edge devices 8. This collaboration simplifies the often complex process of moving models from the flexible PyTorch environment to the resource-constrained edge. The typical workflow involves converting a trained PyTorch model into a **quantized TOSA representation** 8 using PyTorch's dynamo export flow. This intermediate TOSA graph is then fed into the ARM Ethos-U compiler, Vela, which has been specifically enhanced with a **TOSA front-end** 8 to understand and process TOSA. Vela then generates the highly optimized machine instructions (command stream) necessary for efficient execution on the Ethos-U family of NPUs, including the U55, U65, and the high-performance U85 8. The ExecuTorch tutorial demonstrates this process using scripts like aot\_arm\_compiler.py to export PyTorch models (e.g., SoftmaxModule, AddModule, MobileNetV2) into .pte files, which can be simulated on the Corstone-300 FVP. While specific examples for RNNs and Transformers are not extensively detailed, the general workflow is applicable provided the operators are supported by the Ethos-U backend.

**TensorFlow Integration:** TensorFlow Lite is a widely adopted framework for deploying machine learning models on edge devices, including ARM-based systems 1. TOSA is intended to be a key target for lowering operators from both TensorFlow and TensorFlow Lite 1, facilitating efficient execution on ARM hardware. The Arm NN SDK is a crucial component in this integration, enabling the execution of TensorFlow Lite models on ARM CPUs and Mali GPUs 18, B\_B3]. Arm NN includes a TOSA reference implementation (TosaRef) and supports the mapping of various TensorFlow Lite operators to TOSA operations 18, B\_B3], suggesting an internal utilization of TOSA for optimization. To help developers ensure the portability of their TensorFlow Lite models, the tosa-checker tool is available 1, allowing them to verify the compatibility of their models with the TOSA specification.

**ONNX Integration:** ONNX Runtime, a high-performance inference engine for ONNX models, provides an Arm NN Execution Provider 27, B\_B4]. This allows ONNX Runtime to leverage the Arm NN SDK for accelerating ONNX models on ARM devices 27. While the direct execution flow might use Arm NN's internal optimizations, there is potential for ONNX models to be converted to TOSA, especially when targeting ARM NPUs like the Ethos-U series 1. The MLIR compiler infrastructure is actively being used to develop legalization passes that can translate ONNX graphs into TOSA 1. The ONNX Runtime documentation provides C/C++ examples for registering the Arm NN execution provider 35, B\_B4].

### TOSA and Neural Network Architectures on ARM

TOSA's design and the supporting software ecosystem enable the efficient deployment of various neural network architectures on ARM platforms.

**Convolutional Neural Networks (CNNs):** CNNs, essential for image processing, are well-supported on ARM. Models like MobileNetV1 and MobileNetV2 can be deployed using TensorFlow Lite with the Arm NN SDK 28, B\_B13] and PyTorch with ExecuTorch and Vela. These frameworks abstract away the complexities of hardware interaction, potentially using TOSA internally for optimization. The ExecuTorch tutorial and the Arm NN SDK documentation 28, B\_B13] provide conceptual examples and steps for deploying CNNs on ARM.

**Recurrent Neural Networks (RNNs):** The TOSA specification includes operators necessary for representing RNNs, including control flow operators like WHILE\_LOOP 4, B\_B1]. While end-to-end deployment examples specifically using TOSA followed by Vela compilation might be less common in the provided snippets, frameworks like TensorFlow Lite for Microcontrollers demonstrate the deployment of RNNs on ARM Cortex-M devices 20, indicating underlying support that could align with TOSA principles.

**Transformer Networks:** The ARM Ethos-U85 NPU stands out with its native support for transformer-based models through its optimized TOSA implementation 7. This makes it highly suitable for running large language models and other advanced AI workloads at the edge. PyTorch-based Transformer models can be deployed to Ethos-U85 using ExecuTorch, which converts them to TOSA 8. ONNX models might leverage ONNX Runtime and Arm NN, with potential future pathways involving TOSA conversion via MLIR 1.

### TOSA as a Research Topic and its Current Applications

TOSA remains an active area of research and development, particularly within the domain of edge AI 1. Ongoing research focuses on expanding the operator set and enhancing the efficiency of implementations on diverse ARM hardware 1. The ML Platform actively hosts discussions and welcomes contributions to the TOSA specification and its reference implementation 2, fostering a collaborative research environment.

The ARM Ethos-U85's native support for TOSA is a significant development for deploying Large Language Models (LLMs) at the edge 8. Optimization techniques like quantization are crucial for running these demanding models efficiently on ARM 11. Projects like ExecuTorch facilitate the deployment of PyTorch-based LLMs on ARM using TOSA 8. While dedicated research papers on "TOSA ARM large language models" are limited in the provided snippets, the general trend of deploying LLMs on ARM using TOSA-enabled hardware is evident 16.

Regarding Large Image Diffusion Models, the provided material does not contain explicit research focusing on "TOSA ARM diffusion models". However, the principles of TOSA enabling efficient deployment of complex models on ARM could potentially extend to diffusion models as well.

### Career Prospects in TOSA and ARM Machine Learning

The demand for expertise in Edge AI and ARM-based machine learning is substantial and growing 6. ARM's pervasive presence in edge devices makes it a central platform for machine learning innovation. Career opportunities include roles such as Machine Learning Engineers, AI Engineers, and Embedded Software Engineers 68. While specific "TOSA in Machine Learning" certifications are not explicitly detailed in the provided material, a strong understanding of TOSA and related tools like Vela would be a valuable asset in this career path.

### Pathways to Excelling in TOSA and ARM Machine Learning

To excel in this field, a combination of essential skills and continuous learning is necessary. Key skills include a strong foundation in machine learning and deep learning, proficiency in Python and C++, understanding of embedded systems and ARM architecture, experience with frameworks like PyTorch, TensorFlow, and ONNX, familiarity with ARM-specific tools (ExecuTorch, Vela, Arm NN SDK), and knowledge of model quantization and optimization 3.

Recommended learning resources include the official TOSA specification, ARM Developer resources, online courses (e.g., Coursera, edX) focusing on edge AI on ARM, and relevant research papers 3, B\_B21, B\_B221.

To enhance skills, engaging in hands-on projects, contributing to open-source initiatives, participating in online forums, and staying updated with the latest research are highly recommended.

### Conclusions

The Tensor Operator Set Architecture (TOSA) plays a crucial role in enabling efficient and portable machine learning on ARM platforms. By providing a standardized intermediate representation, TOSA bridges the gap between high-level machine learning frameworks and the diverse range of ARM hardware, including CPUs, GPUs, and specialized NPUs like the Ethos-U family. This is particularly significant for the rapidly growing field of Edge AI, where ARM processors are widely used. The integration of TOSA with popular frameworks like PyTorch, TensorFlow, and ONNX, facilitated by tools like ExecuTorch, Arm NN SDK, and the Vela compiler, streamlines the deployment process for various neural network architectures, including CNNs, RNNs, and Transformers. As a research topic, TOSA continues to evolve, with ongoing efforts to expand its capabilities and optimize its performance. Its application in emerging areas like large language models on ARM demonstrates its potential to power the next generation of intelligent edge devices. For individuals looking to build a career in this exciting domain, acquiring a strong foundation in machine learning, proficiency in relevant programming languages, and specialized knowledge of TOSA and ARM-specific tools will be highly advantageous. Continuous learning and engagement with the active TOSA and ARM machine learning communities are essential for staying at the forefront of this dynamic field.

| **Category** | **Description** | **Examples** |
| --- | --- | --- |
| Tensor Operators | Core computational operations on tensors. | CONV2D, MATMUL, AVG\_POOL2D, MAX\_POOL2D |
| Activation Functions | Non-linear functions applied element-wise. | SIGMOID, TANH, RELU |
| Elementwise Operators | Operations performed element-wise on tensors. | ADD, SUB, MUL, DIV, ABS, NEGATE |
| Reduction Operators | Operations that reduce the dimensionality of a tensor. | REDUCE\_SUM, REDUCE\_MAX, REDUCE\_MIN |
| Data Layout Operators | Operations that manipulate the shape or arrangement of tensors. | RESHAPE, TRANSPOSE, SLICE, CONCAT |
| Control Flow Operators | Operators that define the flow of execution in a graph. | COND\_IF, WHILE\_LOOP |
| Other Operators | Includes scatter/gather, image, type conversion, data nodes, custom, variable, and shape operators. | GATHER, RESIZE, CAST, CONST, VARIABLE |

| **NPU Model** | **MAC Units** | **Peak Performance (at 1 GHz)** | **Target Applications** | **Transformer Support** |
| --- | --- | --- | --- | --- |
| Ethos-U55 | Up to 256 | Up to 0.5 TOPs | Cortex-M based systems | Limited |
| Ethos-U65 | Up to 512 | Up to 1 TOPs | Cortex-A based systems | Limited |
| Ethos-U85 | Up to 2048 | Up to 4 TOPs | Cortex-A and Cortex-M based systems | Native |

| **Framework** | **Integration Tool/Library** | **Workflow** | **Notes** |
| --- | --- | --- | --- |
| PyTorch | ExecuTorch, Vela compiler | PyTorch Model -> Quantized TOSA -> Vela -> ARM Ethos-U NPU | Simplifies deployment to ARM edge devices |
| TensorFlow | Arm NN SDK, TOSA Checker | TensorFlow Lite Model -> Arm NN (potentially leveraging TOSA) -> ARM CPU/GPU/NPU | Arm NN provides TOSA reference implementation |
| ONNX | ONNX Runtime, Arm NN Execution Provider, MLIR | ONNX Model -> ONNX Runtime (Arm NN backend) -> ARM CPU/GPU/NPU; Potential ONNX -> TOSA -> Vela pathway | MLIR facilitates ONNX to TOSA conversion |

| **Skill** | **Learning Resources** |
| --- | --- |
| Machine Learning Fundamentals | Online courses (Coursera, edX), textbooks |
| Deep Learning | Specializations on Coursera (e.g., DeepLearning.AI), online courses |
| Python Programming | Python.org, online tutorials, coding platforms |
| C++ Programming | CppReference.com, online courses, textbooks |
| Embedded Systems | Textbooks, online courses (e.g., Arm Education), development boards |
| ARM Architecture | ARM Developer website, processor datasheets, online resources |
| PyTorch | PyTorch official documentation, tutorials, online courses |
| TensorFlow | TensorFlow official documentation, tutorials, online courses |
| ONNX | ONNX official documentation, tutorials |
| ExecuTorch | ExecuTorch documentation on pytorch.org, GitHub repository |
| Vela Compiler | Vela documentation on mlplatform.org, ARM Developer website |
| Arm NN SDK | Arm NN SDK documentation on GitHub, ARM Developer website |
| Model Quantization & Optimization | Research papers, online articles, framework documentation |

#### Works cited

1. Arm Tech Talk: Introducing a common operator set for Machine Learning accelerators – TOSA - YouTube, accessed on March 17, 2025, <https://www.youtube.com/watch?v=I-mljwralfU>
2. Tensor Operator Set Architecture (TOSA) - Arm Community, accessed on March 17, 2025, <https://community.arm.com/cfs-file/__key/communityserver-blogs-components-weblogfiles/00-00-00-37-98/Eric-Kunze-_2D00_-ODIW-2021-_2D00_.pdf>
3. Tensor Operator Set Architecture (TOSA) - Machine Learning Platform, accessed on March 17, 2025, <https://www.mlplatform.org/tosa/>
4. TOSA 1.0.0 draft specification - Machine Learning Platform, accessed on March 17, 2025, <https://www.mlplatform.org/tosa/tosa_spec.html>
5. Tensor Operator Set Architecture (TOSA) Dialect - MLIR - LLVM, accessed on March 17, 2025, <https://mlir.llvm.org/docs/Dialects/TOSA/>
6. Arm Ethos-U85 NPU Technical Overview, accessed on March 17, 2025, <https://developer.arm.com/documentation/102684/0000/Description-of-the-Arm--Ethos-U85--NPU-->
7. Arm Ethos-U NPU - Arm Developer, accessed on March 17, 2025, <https://developer.arm.com/documentation/109267/latest/Arm-Ethos-U-NPU>
8. Getting started with PyTorch, ExecuTorch, and Ethos-U85 in three ..., accessed on March 17, 2025, <https://pytorch.org/blog/pt-executorch-ethos-u85/>
9. Ethos-U85 | Advanced NPU with Scalable Performance and Efficiency – Arm®, accessed on March 17, 2025, <https://www.arm.com/products/silicon-ip-cpu/ethos/ethos-u85>
10. Getting Started With ExecuTorch And Ethos-U85 - AI blog - Arm ..., accessed on March 17, 2025, <https://community.arm.com/arm-community-blogs/b/ai-blog/posts/executorch-support-ethos-u85>
11. ExecuTorch and TOSA enabling PyTorch on Arm platforms - AI blog, accessed on March 17, 2025, <https://community.arm.com/arm-community-blogs/b/ai-blog/posts/executorch-and-tosa-enabling-pytorch-on-arm-platforms>
12. Building and Running ExecuTorch with ARM Ethos-U Backend ..., accessed on March 17, 2025, <https://pytorch.org/executorch/0.3/executorch-arm-delegate-tutorial.html>
13. Arm Collaborates with Industry Leaders to Build AI Foundations of the Future, accessed on March 17, 2025, <https://www.design-reuse.com/news/55124/arm-amd-intel-meta-microsoft-nvidia-qualcomm-collaboration.html>
14. kyoshidajp/tosa: Open pull request page from commit hash tool - GitHub, accessed on March 17, 2025, <https://github.com/kyoshidajp/tosa>
15. llvm-project/mlir/docs/Dialects/TOSA.md at main - GitHub, accessed on March 17, 2025, <https://github.com/llvm/llvm-project/blob/main/mlir/docs/Dialects/TOSA.md>
16. AI blog - Arm Community, accessed on March 17, 2025, <https://community.arm.com/arm-community-blogs/b/ai-blog>
17. AI Software Development Platform | Arm Solutions, accessed on March 17, 2025, <https://www.arm.com/markets/artificial-intelligence/software>
18. Releases · ARM-software/armnn - GitHub, accessed on March 17, 2025, <https://github.com/ARM-software/armnn/releases>
19. ethos-u-vela · PyPI, accessed on March 17, 2025, <https://pypi.org/project/ethos-u-vela/>
20. GitHub - nxp-imx/ethos-u-vela, accessed on March 17, 2025, <https://github.com/nxp-imx/ethos-u-vela>
21. Vela compiler - ML Developers Guide for Cortex-M Processors and Ethos-U NPU, accessed on March 17, 2025, <https://developer.arm.com/documentation/109267/latest/Tool-support-for-the-Arm-Ethos-U-NPU/Ethos-U-Vela-compiler>
22. ethos-u-vela/OPTIONS.md at lf-6.6.3\_1.0.0 - GitHub, accessed on March 17, 2025, <https://github.com/nxp-imx/ethos-u-vela/blob/lf-6.6.3_1.0.0/OPTIONS.md>
23. AN13854: NPU Migration Guide from i.MX 8M Plus to i.MX 93 | NXP Semiconductors, accessed on March 17, 2025, <https://docs.nxp.com/bundle/AN13854/page/topics/vela_tool.html>
24. tinyML Development with Tensorflow Lite using CMSIS-NN and Ethos-U55, accessed on March 17, 2025, <https://developer.arm.com/Additional%20Resources/Video%20Tutorials/AITechTalk-Video-tinyML%20Development%20with%20Tensorflow%20Lite%20using%20CMSIS-NN%20and%20Ethos-U55>
25. Arm NN: the Easy Way to Deploy Edge ML - Tools, Software and IDEs blog, accessed on March 17, 2025, <https://community.arm.com/arm-community-blogs/b/tools-software-ides-blog/posts/arm-nn-the-easy-way-to-deploy-edge-ml>
26. Accelerated inference on Arm microcontrollers with TensorFlow Lite for Microcontrollers and CMSIS-NN, accessed on March 17, 2025, <https://blog.tensorflow.org/2021/02/accelerated-inference-on-arm-microcontrollers-with-tensorflow-lite.html>
27. Arm NN, Compute Library and Arm MLIA - ML PLatform, accessed on March 17, 2025, <https://www.mlplatform.org/>
28. developer.arm.com, accessed on March 17, 2025, <https://developer.arm.com/-/media/Arm%20Developer%20Community/PDF/Machine%20Learning/Machine%20Learning%20PDF%20Tutorials/Deploying%20a%20quantized%20TensorFlow%20Lite%20MobileNet%20V1%20model%20using%20the%20Arm%20NN%20SDK.pdf?revision=81bba3e8-9e3a-4653-bb50-80e2c6c210b1>
29. AI Tech Talk: Optimizing NN inference performance on Arm NEON and Vulkan using the Ailia SDK - YouTube, accessed on March 17, 2025, <https://www.youtube.com/watch?v=lUmjnCdGtGE>
30. Optimizing Machine Learning Workloads on Power-efficient Devices - NET, accessed on March 17, 2025, <https://armkeil.blob.core.windows.net/developer/Files/pdf/white-paper/optimizing-machine-learning-workloads.pdf>
31. Arm NN SDK | Efficient ML for Arm CPUs, GPUs, & NPUs, accessed on March 17, 2025, <https://www.arm.com/products/silicon-ip-cpu/ethos/arm-nn>
32. Optimizing NN inference performance on Arm NEON and Vulkan using ailia SDK, accessed on March 17, 2025, <https://developer.arm.com/Additional%20Resources/AITechTalk-Slides-Optimizing%20NN%20inference%20performance%20on%20Arm%20NEON%20and%20Vulkan%20using%20the%20Ailia%20SDK>
33. How to Build Optimized ML Applications with Arm Software - Hubilo, accessed on March 17, 2025, <https://cdn.hubilo.com/brochure/10860/3466_8328_913525001543475480.pdf>
34. tosa-checker - PyPI, accessed on March 17, 2025, <https://pypi.org/project/tosa-checker/>
35. Arm - Arm NN | onnxruntime, accessed on March 17, 2025, <https://onnxruntime.ai/docs/execution-providers/community-maintained/ArmNN-ExecutionProvider.html>
36. Deploy Transformer Models in the Browser with #ONNXRuntime - YouTube, accessed on March 17, 2025, <https://www.youtube.com/watch?v=W_lUGPMW_Eg>
37. Transformer Models To Onnx Conversion | Restackio, accessed on March 17, 2025, <https://www.restack.io/p/transformer-models-answer-onnx-conversion-cat-ai>
38. Transformers optimizer | onnxruntime, accessed on March 17, 2025, <https://onnxruntime.ai/docs/performance/transformers-optimization.html>
39. transformers-inference-experiments/onnx/simple\_mrpc\_example.ipynb at main - GitHub, accessed on March 17, 2025, <https://github.com/philschmid/transformers-inference-experiments/blob/main/onnx/simple_mrpc_example.ipynb>
40. Convert Transformers to ONNX with Hugging Face Optimum - Philschmid, accessed on March 17, 2025, <https://www.philschmid.de/convert-transformers-to-onnx>
41. I want insights and experiences related to exploring the TOSA ..., accessed on March 17, 2025, <https://discuss.mlplatform.org/t/i-want-insights-and-experiences-related-to-exploring-the-tosa-specification/311>
42. EasyChair Preprint SMT-Based Translation Validation for Machine Learning Compiler, accessed on March 17, 2025, <https://easychair.org/publications/preprint/LjP7/open>
43. Products and applications enabled by tinyML, accessed on March 17, 2025, <https://cms.tinyml.org/wp-content/uploads/ew2023/Ian-Bratt_tinyML-Summit.pdf>
44. Forums - ML PLatform, accessed on March 17, 2025, <https://www.mlplatform.org/forums/>
45. Discourse, accessed on March 17, 2025, <https://discuss.mlplatform.org/>
46. Harnessing the Power of LLM Models on Arm CPUs for Edge Devices, accessed on March 17, 2025, <https://www.edge-ai-vision.com/2025/01/harnessing-the-power-of-llm-models-on-arm-cpus-for-edge-devices/>
47. Run a Large Language model (LLM) chatbot on Arm servers - Arm Learning Paths, accessed on March 17, 2025, <https://learn.arm.com/learning-paths/servers-and-cloud-computing/llama-cpu/llama-chatbot/>
48. Deploy a Large Language Model (LLM) chatbot with llama.cpp ..., accessed on March 17, 2025, <https://learn.arm.com/learning-paths/servers-and-cloud-computing/llama-cpu/>
49. Enabling Generative AI at Scale - Arm, accessed on March 17, 2025, <https://www.arm.com/markets/artificial-intelligence/generative-ai>
50. Large Language Diffusion Models (LLaDA) Explained - New Generative AI Paradigm, accessed on March 17, 2025, <https://www.youtube.com/watch?v=aT5S3MZcWJA>
51. arxiv.org, accessed on March 17, 2025, <https://arxiv.org/abs/2502.09992>
52. All-atom Diffusion Transformers: Unified generative modelling of ..., accessed on March 17, 2025, <https://www.arxiv.org/abs/2503.03965>
53. Deploy Diffusion Models on Lambdas - Nicolas Bertagnolli, accessed on March 17, 2025, <http://www.nbertagnolli.com/jekyll/update/2024/07/01/deploy_diffusion_models_to_lambda.html>
54. Understanding Diffusion Models: An Essential Guide for AEC Professionals, accessed on March 17, 2025, <https://developer.nvidia.com/blog/understanding-diffusion-models-an-essential-guide-for-aec-professionals/>
55. How to install and run Stable Diffusion on Apple Silicon M1/M2 Macs, accessed on March 17, 2025, <https://stable-diffusion-art.com/install-mac/>
56. Diffusion Models: A Practical Guide - Scale AI, accessed on March 17, 2025, <https://scale.com/guides/diffusion-models-guide>
57. ML Acceleration Microcontroller Subsystems - Arm, accessed on March 17, 2025, <https://www.arm.com/resources/education/academic-developer-platforms/ml-acceleration-microcontroller-subsystems>
58. AI Technologies | Products, Solutions and Software for Any AI Workload - Arm, accessed on March 17, 2025, <https://www.arm.com/markets/artificial-intelligence/technologies>
59. Machine Learning - Artificial Intelligence – Arm, accessed on March 17, 2025, <https://www.arm.com/markets/artificial-intelligence/machine-learning>
60. Arm Server Market Size, Growth, Trends, Report 2034, accessed on March 17, 2025, <https://www.marketresearchfuture.com/reports/arm-server-market-26445>
61. ARM Microcontroller Market Share Report, Outlook, Growth 2034, accessed on March 17, 2025, <https://www.marketresearchfuture.com/reports/arm-microcontroller-market-32787>
62. Arm Tech Predictions for 2025 and Beyond, accessed on March 17, 2025, <https://newsroom.arm.com/blog/arm-2025-tech-predictions>
63. Chapter 3: The Technological Enablers of Edge AI - Wevolver, accessed on March 17, 2025, <https://www.wevolver.com/article/2025-edge-ai-technology-report/the-technological-enablers-of-edge-ai>
64. Accelerating Adoption of Edge AI - Supermicro, accessed on March 17, 2025, <https://www.supermicro.com/white_paper/white_paper_Edge_AI_SMCI_NVIDIA.pdf>
65. Survey Reveals Edge AI Adoption in Manufacturing Is Solving Complex Challenges, accessed on March 17, 2025, <https://www.manufacturingdive.com/press-release/20240920-survey-reveals-edge-ai-adoption-in-manufacturing-is-solving-complex-challen-1/>
66. Edge AI: Adoption, Key Players, and Outlook - AlphaSense, accessed on March 17, 2025, <https://www.alpha-sense.com/blog/trends/edge-ai-expert-insights-dor/>
67. As AI nurses reshape hospital care, human nurses are pushing back - AP News, accessed on March 17, 2025, <https://apnews.com/article/artificial-intelligence-ai-nurses-hospitals-health-care-3e41c0a2768a3b4c5e002270cc2abe23>
68. $32k-$230k Edge Ai Machine Learning Jobs (NOW HIRING) Mar 25 - ZipRecruiter, accessed on March 17, 2025, <https://www.ziprecruiter.com/Jobs/Edge-Ai-Machine-Learning>
69. Edge AI Engineering Roles Overview | Restackio, accessed on March 17, 2025, <https://www.restack.io/p/edge-ai-answer-engineering-roles-cat-ai>
70. Senior Machine Learning Engineer, Edge AI - Remote - US - Samsara, accessed on March 17, 2025, <https://www.samsara.com/ca/company/careers/roles/6569967?gh_jid=6569967>
71. Edge AI - Arm, accessed on March 17, 2025, <https://www.arm.com/markets/iot/edge-ai>
72. What is Edge AI? - Arm, accessed on March 17, 2025, <https://www.arm.com/glossary/edge-ai>
73. Arm machine learning engineer jobs - ZipRecruiter, accessed on March 17, 2025, <https://www.ziprecruiter.com/Jobs/Arm-Machine-Learning-Engineer>
74. Software Engineer in Machine Learning - Arm Careers, accessed on March 17, 2025, <https://careers.arm.com/job/galway/software-engineer-in-machine-learning/33099/69202964112>
75. Arm Graduate Machine Learning Engineer Interview Questions - NodeFlair, accessed on March 17, 2025, <https://nodeflair.com/companies/arm/interviews/graduate-machine-learning-engineer>
76. Explore Career Paths on Microsoft Learn, accessed on March 17, 2025, <https://learn.microsoft.com/en-us/training/career-paths/>
77. Arm Machine Learning ArmNN - Arm Developer, accessed on March 17, 2025, <https://developer.arm.com/Training/Arm%20Machine%20Learning%20ArmNN>
78. Arm Job Search Results for Machine Learning - Arm Careers, accessed on March 17, 2025, <https://careers.arm.com/category/machine-learning-jobs/33099/8110432/1>
79. IoT Software Product Owner - Arm Careers, accessed on March 17, 2025, <https://careers.arm.com/job/sophia-antipolis/iot-software-product-owner/33099/77089001104>
80. Working at Arm | Jobs & Careers, accessed on March 17, 2025, <https://careers.arm.com/>
81. Arm Ethos-U Linux Driver Posted For Machine Learning Processor - Phoronix, accessed on March 17, 2025, <https://www.phoronix.com/news/Arm-Ethos-U-Linux-Driver>
82. Machine Learning Skills: Your Guide to Getting Started | Coursera, accessed on March 17, 2025, <https://www.coursera.org/articles/machine-learning-skills>
83. Getting Started with Machine Learning at the Edge on Arm | Coursera, accessed on March 17, 2025, <https://www.coursera.org/learn/getting-started-with-machine-learning-at-the-edge-on-arm>
84. ArmEducationX: Machine Learning at the Edge on Arm: A Practical ..., accessed on March 17, 2025, <https://www.edx.org/learn/machine-learning/arm-education-machine-learning-at-the-edge-on-arm-a-practical-introduction>
85. ML Developers Guide for Cortex-M Processors and Ethos-U NPU, accessed on March 17, 2025, <https://developer.arm.com/documentation/109267/latest/Arm-Ethos-U-NPU/Ethos-U-hardware-architecture>
86. TensorFlow Tutorial 10 - Recurrent Neural Nets (RNN & LSTM & GRU) - YouTube, accessed on March 17, 2025, <https://www.youtube.com/watch?v=IrPhMM_RUmg>
87. Tensorflow RNN examples - by Clement Joudet - Medium, accessed on March 17, 2025, <https://medium.com/@clemkoa/tensorflow-rnn-road-to-glory-f52967918a2c>
88. Unrolling RNN models for Arm Ethos-U55 and U65 - ARM Community Forums, accessed on March 17, 2025, <https://community.arm.com/arm-community-blogs/b/ai-blog/posts/rnn-models-ethos-u>
89. Working with RNNs | TensorFlow Core, accessed on March 17, 2025, <https://www.tensorflow.org/guide/keras/working_with_rnns>
90. PyTorch Examples — PyTorchExamples 1.11 documentation, accessed on March 17, 2025, <https://pytorch.org/examples/>
91. Quantize ONNX models | onnxruntime, accessed on March 17, 2025, <https://onnxruntime.ai/docs/performance/model-optimizations/quantization.html>
92. Parsers - Arm NN - GitHub Pages, accessed on March 17, 2025, <https://arm-software.github.io/armnn/24.02/parsers.html>
93. Quantize ONNX Models - ONNXRuntime - GitHub Pages, accessed on March 17, 2025, <https://iot-robotics.github.io/ONNXRuntime/docs/performance/quantization.html>
94. Building and Running ExecuTorch with ARM Ethos-U Backend - PyTorch, accessed on March 17, 2025, <https://pytorch.org/executorch/stable/executorch-arm-delegate-tutorial.html>
95. ML Developers Guide for Cortex-M Processors and Ethos-U NPU, accessed on March 17, 2025, <https://developer.arm.com/documentation/109267/latest/Tool-support-for-the-Arm-Ethos-U-NPU/Ethos-U-Vela-compiler/Optimization-considerations-for-the-Vela-compiler>
96. Accelerating AI on MaaXBoard OSM93 - Part 2: Vela Conversion - Hackster.io, accessed on March 17, 2025, <https://www.hackster.io/monica/accelerating-ai-on-maaxboard-osm93-part-2-vela-conversion-0e700a>
97. Developer Program – Arm®, accessed on March 17, 2025, <https://www.arm.com/resources/developer-program>
98. Introducing TOSA: A Common Operator Set for Machine Learning Accelerators, accessed on March 17, 2025, <https://www.classcentral.com/course/youtube-arm-tech-talk-introducing-a-common-operator-set-for-machine-learning-accelerators-tosa-319953>
99. mlir/docs/Dialects/TOSA.md · plugins-review-02 · Kevin Sala / llvm-project - GitLab, accessed on March 17, 2025, <https://git.cels.anl.gov/ksalapenades/llvm-project/-/blob/plugins-review-02/mlir/docs/Dialects/TOSA.md>
100. Take your neural networks to the next level with Arm's Machine Learning Inference Advisor - Arm Community, accessed on March 17, 2025, <https://community.arm.com/arm-community-blogs/b/ai-blog/posts/arm-machine-learning-inference-advisor>
101. EML TOSA Market's Growth Blueprint, accessed on March 17, 2025, <https://www.datainsightsmarket.com/reports/eml-tosa-1671181>
102. ML Developers Guide for Cortex-M Processors and Ethos-U NPU, accessed on March 17, 2025, <https://developer.arm.com/documentation/109267/latest/Tool-support-for-the-Arm-Ethos-U-NPU/Machine-Learning-Inference-Advisor>
103. Post-weld-shift compensation strategy with improved U-net edge detection for reducing coupling loss of 25Gbps TOSA, accessed on March 17, 2025, <https://opg.optica.org/oe/viewmedia.cfm?uri=oe-33-6-12852&seq=0>
104. ToSA: A Top-Down Tree Structure Awareness Model for Hierarchical Text Classification - ResearchGate, accessed on March 17, 2025, <https://www.researchgate.net/profile/Shuangyong-Song/publication/368452255_ToSA_A_Top-Down_Tree_Structure_Awareness_Model_for_Hierarchical_Text_Classification/links/640ee030a1b72772e4f22e1a/ToSA-A-Top-Down-Tree-Structure-Awareness-Model-for-Hierarchical-Text-Classification.pdf>
105. Cross-vendor programming abstraction for diverse heterogeneous platforms - Frontiers, accessed on March 17, 2025, <https://www.frontiersin.org/journals/computer-science/articles/10.3389/fcomp.2022.945652/full>
106. (PDF) ToSA: Token Selective Attention for Efficient Vision Transformers - ResearchGate, accessed on March 17, 2025, <https://www.researchgate.net/publication/381404797_ToSA_Token_Selective_Attention_for_Efficient_Vision_Transformers>