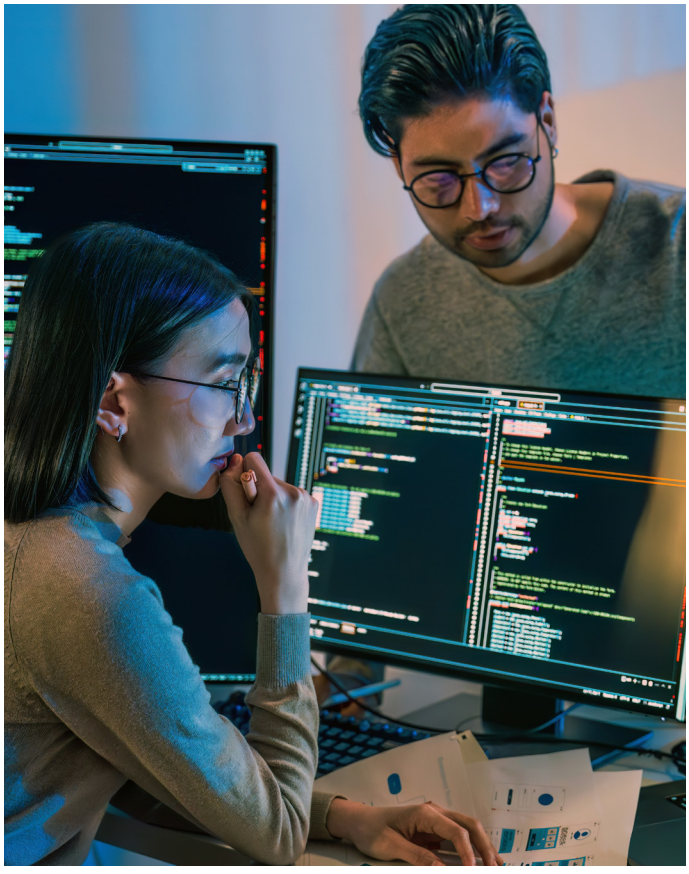
PROWESS

Research Abstract

Optimized Toolkits for LLM Deployment on AI PCs

With the emergence of AI PCs, developers can now run large language models and AI applications locally, leveraging on-device acceleration and hardware-specific optimizations. A Prowess Consulting report evaluates hardware-optimized toolkits designed to deliver high-performance, low-latency AI inference on AI-enabled PCs.

Developers increasingly find themselves on the frontlines of AI software development for enterprise environments. Many developers have an array of AI tools, large language models (LLMs), and hardware environments to choose from as they enhance existing applications and build natural language processing (NLP) capabilities into business processes and related software.



With the transition to AI PCs equipped with on-device hardware acceleration for AI and machine learning (ML) workloads, developers can now securely deploy models in local desktop environments, eliminating the need for external API and server connections. In addition to reducing costs, local development enables more efficient and flexible deployment of LLMs, supporting low-latency on-device inference while ensuring data privacy and compliance with security requirements.

In a report commissioned by Intel, Prowess Consulting examines the benefits and challenges of using AI acceleration and hardware-optimized tooling on Dell™ XPS™ 13 devices, powered by Intel® Core™ Ultra processors and Qualcomm® Snapdragon® Elite Arm64 systems on a chip. We put the open-source Intel® OpenVINO™ toolkit and the Qualcomm® AI Engine Direct SDK to the test, running and optimizing LLMs locally and eliminating the need for cloud services.

Which Tools Are Best for Working with LLMs?

To find out which development environment came out on top, we created an AI pipeline for each hardware-specific software development kit (SDK). We evaluated the effectiveness of the tooling for working with a chatbot powered by the open-source Meta Llama 3.2-3B (3 billion parameter) LLM, including model conversion and inference on both hardware platforms.

We implemented an INT8 quantization workflow using the Intel OpenVINO toolkit and Qualcomm AI Engine Direct SDK and tested the performance against the three types of processors on the respective hardware: general-purpose central processing units (CPUs), parallel processing graphic processing units (GPUs), and AI/ML performance–enhancing neural processing units (NPUs).

We evaluated the benefits of using hardware-optimized tooling for AI and ML workloads on Dell XPS 13 devices, powered by Intel Core Ultra processors and Qualcomm Snapdragon Elite Arm64 SOCs. To test AI inference against each hardware platform's CPU, GPU and NPU, we implemented the Intel OpenVINO toolkit and Qualcomm AI Engine Direct SDK to run LLMs locally.

Developer-Focused Toolkit Ratings

Our findings suggest that one toolkit, which outperformed the other in the majority of our ratings, is the better choice for working with LLMs, as shown in Table 1. We were able to download and convert the pretrained Llama 3.2-3B model to deploy in the toolkit's runtime, perform quantization to INT8 and INT4, and target the desired hardware. With this toolkit, much of this process has been streamlined and can be executed via the command line interface with minimal programming.

Table 1 | SDK scorecard: The Intel® OpenVINO™ toolkit versus the Qualcomm® AI Engine Direct SDK (five-star rating system: 1 [poor] to 5 [excellent])

Software Development Toolkits	Target Hardware	Platform Compatibility	Model Conversion	Inference	Community Support
Intel® Distribution of OpenVINO™ toolkit	★★★★★	★★★★☆	★★★★★	★★★★☆	★★★★★
Qualcomm® AI Engine Direct SDK	★★★☆☆	★★★☆☆	★☆☆☆☆	★★☆☆☆	★★☆☆☆

Primary drivers influencing the SDK ratings in Table 1:

- Less availability of Qualcomm® Snapdragon® X processor–based systems compared to Intel® Core™ Ultra processor–based systems, signaling slower adoption of Arm® architectures and software compatibility issues
- The Qualcomm® AI Engine Direct SDK offers enough features to run full AI pipelines, but some tools have limited functionality, particularly when working with LLMs

Challenges in Quantizing the Model

We could not perform low-precision quantization techniques on Llama 3.2-3B using the Qualcomm AI Engine Direct SDK. Other tasks, such as tracing the model, proved difficult. At the time of this research, Qualcomm AI Hub Quantization was still in beta.[1] While a smaller model could potentially complete the quantization process, our testing ran into roadblocks that prevented it on large models. The LLM could be built using the Qualcomm GenAI and Inference Extensions (GENIE), a software library and framework, but the inference prompt failed to reply as expected.

Prowess Consulting's research focused on optimizing LLMs on Intel x86-64 Core Ultra and Qualcomm Snapdragon X Elite Arm64 processors on Windows PCs, comparing model conversion, inference, and deployment workflows. We evaluated the tools and assessed the challenges of using the Intel OpenVINO toolkit compared to the Qualcomm AI Engine Direct SDK. Our findings indicate that the Intel OpenVINO toolkit provides a more efficient and flexible workflow for developers deploying a broad range of LLMs on AI-enabled PCs.

Learn More

Get the full story by reading the technical research report,
"Which Toolkit Provides the Best Optimization for Large Language Models?"

Endnotes

[1] Qualcomm Technologies, Inc. 2025. Overview of Qualcomm AI Hub, "**Quantization (Beta)**." Accessed May 2025.

Legal Notices and Disclaimers

The analysis in this document was done by Prowess Consulting and commissioned by Intel.

Results have been simulated and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance.

Prowess and the Prowess logo are trademarks of Prowess Consulting, LLC.

Copyright © 2025 Prowess Consulting, LLC. All rights reserved.
Other trademarks are the property of their respective owners.

0725/240157