Accelerating RAG Pipelines for Enterprise LLM Applications using OpenVINO on the Lenovo ThinkSystem SR650 V3 with 5th Gen Intel Xeon Scalable Processors

Solution Brief

Deploy and Scale Generative AI in Enterprises

Generative AI is transforming businesses today as enterprises incorporate automated content generation processes into their workflows to more efficiently meet customer needs. Large Language Models (LLMs) have shown a great ability to understand and generate human-like text, and can be used in many applications, such as customer service, content creation and summarization, healthcare, and entertainment.

Despite having demonstrated remarkable capabilities in natural language processing tasks, LLMs are known to face significant challenges when handling inputs that extend beyond their training data. They require substantial computational resources to be trained or fine-tuned to gain domain- or company-specific knowledge, incorporate up-to-date information, and reduce the chance of hallucinations.

Retrieval Augmented Generation (RAG) is a set of techniques that allow businesses to leverage the generative AI power of LLMs without having to spend all the time and cost required for model training or fine-tuning. Simultaneously, RAG ensures that the generative model uses current and pertinent information, thus enabling the production of accurate, high-quality content that more closely adheres to a company's unique policies and objectives.

This paper presents the impressive capabilities of the OpenVINO toolkit to accelerate RAG pipelines by using the AI acceleration features present in the Lenovo ThinkSystem SR650 V3 with 5th Gen Intel® Xeon® Scalable processors for the optimization of LLMs and vector embedding models. This solution equips businesses to successfully embark on their generative AI journey using LLM-based applications. This propels organizations towards transformative business outcomes without the need for dedicated (and often costly) GPU accelerators. Additionally, it offers a scalable solution that can adapt to increasing demand while meeting the low latency requirements of LLM applications.

Solution Overview

The Lenovo ThinkSystem SR650 V3, with Intel 5th Gen Xeon processors, delivers a highly performant and scalable solution for Generative AI use cases, including those with low-latency requirements for a successful user experience, like real-time chatbots (with target latency of ~100ms). It offers multiple storage and networking options in a single 2U server that adapts to varied business requirements while providing seamless scalability to adapt to changing needs. It supports DDR5-5600 MT/s memory modules, and one or two 5th Gen Intel Xeon processors which incorporate Intel Advanced Matrix Extensions (Intel AMX) to meet the compute-intensive requirements of cutting-edge AI workloads. Furthermore, it contains three drive bay zones that support up to 20x 3.5-inch or 40x 2.5-inch hot swap drive bays for efficient and scalable storage.

In addition to providing exceptional performance for the processing of advanced AI workloads, the ThinkSystem SR650 V3 offers energy-efficiency features to help conserve power and lower operational expenses. These features include advanced direct-water cooling (DWC) with the Lenovo Neptune® Processor DWC Module, where heat from the processors is removed from the rack and data center using an open loop and coolant distribution units, resulting in lower energy costs, high-efficiency power supplies with 80 PLUS Titanium certifications, and optional Lenovo XClarity Energy Manager, which provides advanced data center power notification, analysis, and policy-based management to help achieve lower heat output and reduced cooling needs.

Figure 1. Lenovo ThinkSystem SR650 V3

Results

For our tests we used Langchain to implement a RAG-based question answering (QA) pipeline that uses the Llama2-7B-chat LLM to generate human-like responses based on documents extracted from a Qdrant vector database. The use of OpenVINO in this application is based on the LLM Chatbot Demo notebook created by the OpenVINO team.

Over 300,000 articles were extracted from an XML dump of the English Wikipedia database and transformed into plain text files, each approximately 1MB in size. These files contained the articles' titles, main bodies, and some basic metadata. Leveraging Langchain's text splitters, the plain text files were segmented to generate over a million individual documents, which were then stored in the Qdrant vector database. Figure 2 shows a diagram of the RAG-based QA pipeline implemented for our tests.

For the LLM input prompt, a subset of eight questions were selected from the rag-mini-wikipedia dataset to be used as queries. The rag-mini-wikipedia dataset is a very convenient option for testing RAG-based QA systems because it offers A) a text corpus split into passages, with each passage assigned a unique numeric ID, and B) a list of questions paired with a corresponding short answer along with a list of the passage IDs that are most relevant to each question.

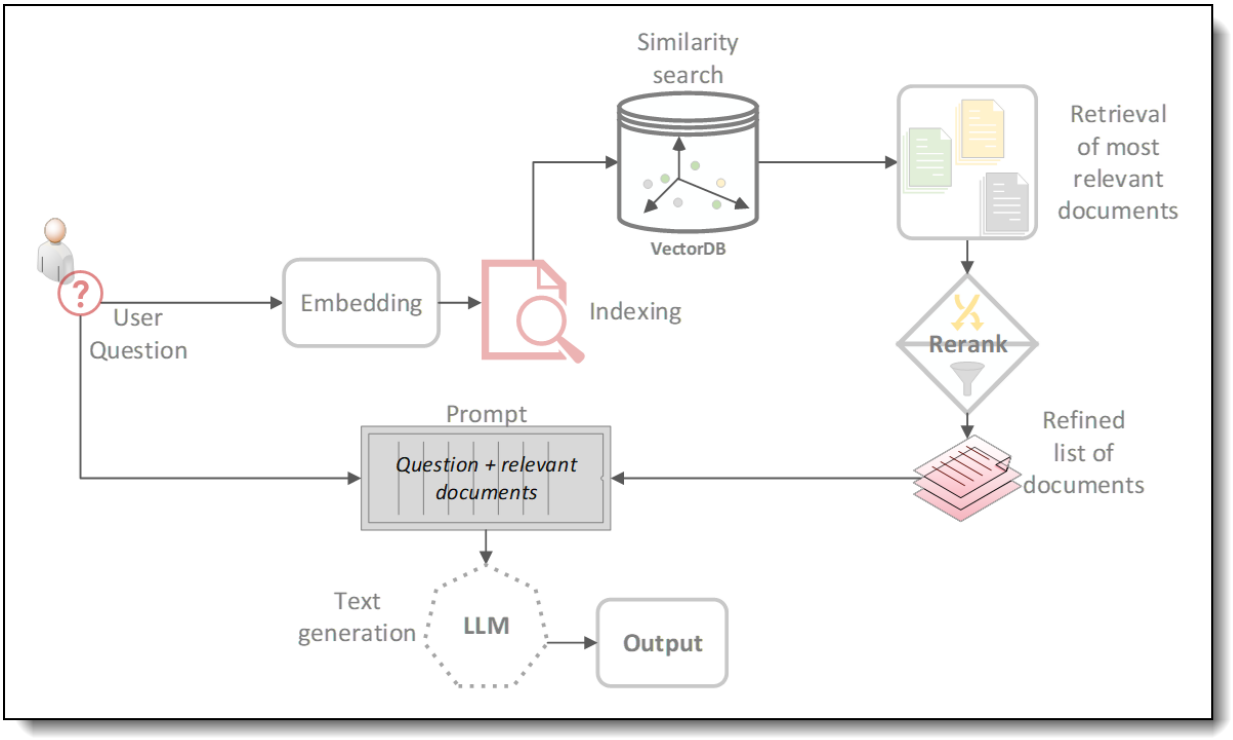

Figure 2. RAG pipeline implemented for QA system

In addition to storing the vector embeddings of the articles, Qdrant was also used as a retrieval tool to fetch the top 10 documents that most closely matched each of the eight input questions in our evaluations. After identifying the top 10 documents most pertinent to each question, we employed the BAAI/bge-reranker-base model to further sift through these results and select the 3 most relevant documents. These documents, paired with the input question, compose the ultimate prompt for the LLM. Each question was processed through the RAG pipeline independently and the first- and next-token latencies recorded.

First-token and next-token: LLMs generate text token by token. The term *first-token latency* refers to the time it takes the LLM to process the user's input prompt and generate the first output token. Similarly, the term *next-token latency* refers to the time it takes (in average) to generate each output token after the first.

The next-token latency is a crucial performance indicator for text-generation tasks that is deemed within an acceptable range if kept under 100ms, aligning with the average human reading pace of 5-10 words per second. For each query we also record the average end-to-end latency which includes the information retrieval and text generation processes.

Figure 3 shows the inference performance speedup obtained with OpenVINO optimizations using BFloat16, INT8, and INT4, compared to Pytorch. The OpenVINO inference is accelerated via the 5th Gen Xeon Scalable Processor built-in AI Acceleration with AMX, and the OpenVINO toolkit with Optimum and Intel's Neural Network Compression Framework (NNCF).

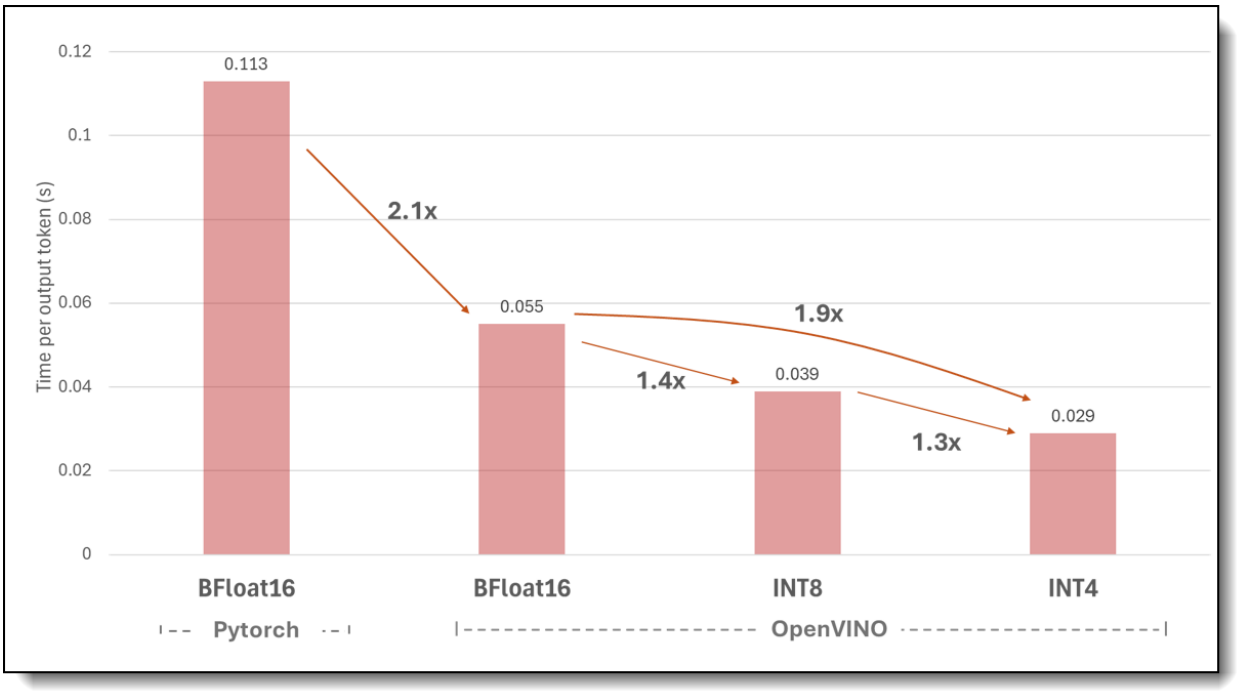

Figure 3. Next-token latency in a RAG pipeline using OpenVINO with 5th Gen Intel Xeon Scalable processor

The following key observations can be made from Figure 3:

- Using OpenVINO allows for a next-token latency that comfortably meets the standard expectations for LLM-based QA systems, maintaining a rate significantly below 100 milliseconds [8].
- Utilizing OpenVINO can lead to performance enhancements, achieving up to a 2.1x increase in speed relative to Pytorch when BF16 precision is used.
- Further acceleration is achievable with OpenVINO using quantization techniques, with INT8 and INT4 quantizations delivering speed boosts of 1.4x and 1.9x, respectively.

In our tests, the first-token latency—which is typically higher and more compute intensive than the next-token latency— consistently remained below 0.8 seconds with the application of *OpenVINO*. On the other hand, when utilizing *Pytorch*, the average latency for the first token was recorded at 1.7 seconds. In addition, Figure 4 shows that a 2x speedup can be obtained when using *OpenVINO* to build the vector database compared to the time required using Pytorch (Hugging Face embedding model can be supported by OpenVINO through OpenVINOEmbeddings class)

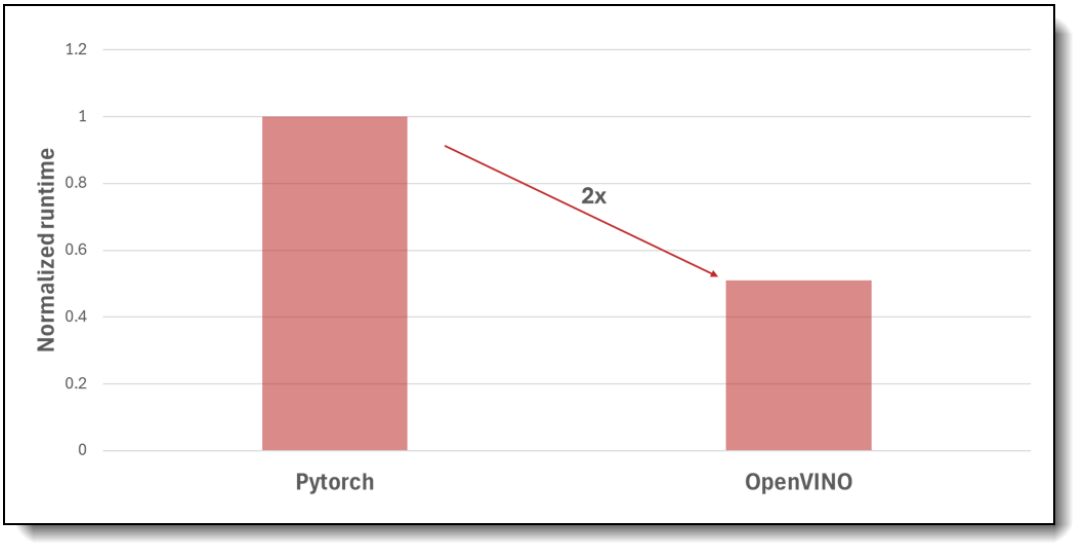

Figure 4. Normalized time required to build vector DB with more than a million documents extracted from Wikipedia

Figure 5 shows the normalized end-to-end runtimes, comprising the processes of retrieval, reranking, and text generation within the RAG pipeline. On average, a 1.6x acceleration is achieved with OpenVINO BF16 in comparison to Pytorch. Moreover, employing INT8 and INT4 model quantizations can provide further performance speedups of 1.2x and 1.3x, respectively

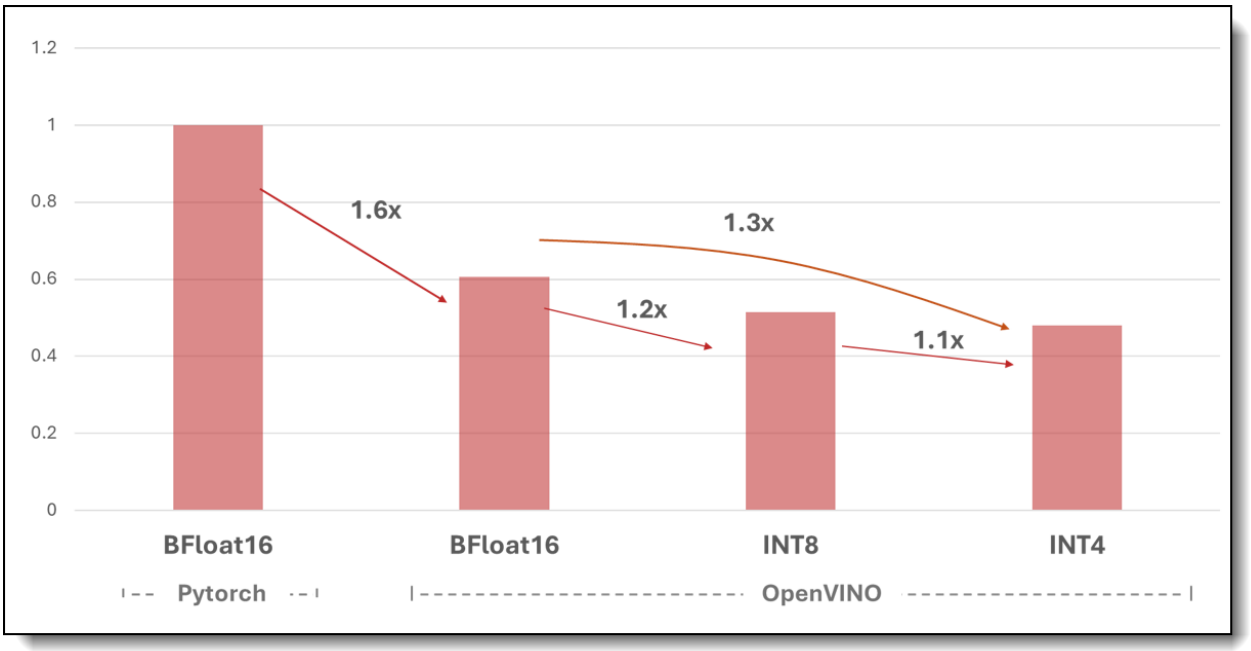

Figure 5. Normalized average end-to-end runtime of QA system. This time includes the time required in average to run through the full RAG pipeline, which involves retrieval, reranking, and text generation phases. While varying queries and data types might yield a diverse count of output tokens, our tests demonstrate that, on average, OpenVINO consistently outpaces Pytorch by a minimum factor of 1.6x.

Configuration Details

The following table lists the configuration of the ThinkSystem SR650 V3 server used in the solution.

Table 1. ThinkSystem SR650 V3 server configuration

Component	Description
Hardware Configuration	
Server	Lenovo ThinkSystem SR650 V3
Processor	Intel Xeon Platinum 8562Y+ processor
Microarchitecture	EMR_MCC
Sockets	2
Cores per Socket	32
Hyperthreading	Intel Hyper-Threading Technology Enabled
CPUs	128
Turbo	Intel Turbo Boost Technology Enabled
Base Frequency	2.8GHz
All-core Maximum Frequency	3.8GHz
Maximum Frequency	4.1GHz
NUMA Nodes	2
Installed Memory	512GB (16x32GB DDR5 5600 MT/s [5600 MT/s])
NIC	2x Ethernet Controller E810-XXV for SFP, 4x I350 Gigabit Network Connection, 2x Ethernet Controller E810-C for QSFP
Disk	2x 894.3G Micron_7450_MTFDKBA960TFR, 1x 2.9T INTEL SSDPF2KE032T1O
BIOS	ESE124B-3.11
Microcode	0x21000200
OS	Red Hat Enterprise Linux CoreOS 414.92.202312011602-0 (Plow)
Kernel	5.14.0-284.43.1.el9_2.x86_64
Software Configuration	
SW versions	openvino-nightly 2024.2.0.dev20240412 langchain 0.1.17 langchain-community 0.0.37 nncf 2.9.0 optimum-intel 1.17.0.dev0+0540b12 qdrant-client 1.8.2 transformers 4.38.2 Python 3.10.14 Datasets 2.18.0 Wheel 0.43.0 Git 2.39.3 GCC 8.5.0 Findutils 4.6.0 Sentencepiece 0.2.0 sentence-transformers 2.5.1 reader 3.12
LLM Models	Meta Llama-2-7b-hf
Dataset	Rag-mini-wikipedia, Wikipedia dump (800MB)

Component	Description
Batch size	1
Max output token size	256
Precision	BF16, INT8, INT4
Input data processing	Langchain's RecursiveCharacterTextSplitter with 1000 chunk size and 100 overlap
Beam Width	1 (greedy search)
Vector DB distance metric	Cosine similarity
Retriever top documents	10
Reranker top documents	3
Vector DB - number of edges per node in the index graph (m)	16
Vector DB - Number of neighbors to consider during the index building (ef)	11

Conclusion

This document showcases the efficiency of OpenVINO in harnessing the AI acceleration capabilities of the Lenovo ThinkSystem SR650 V3 with 5th Gen Intel Xeon Scalable processors within an LLM-based RAG pipeline. Our results indicate that utilization can lead to more than 2x improvement in next-token latency compared to Pytorch with BF16 precision. Additionally, model quantization with OpenVINO can yield up to a 1.9x increase in speed. We also show that usage can enhance first-token latency by more than 2.1x as well as provide a 2x speedup in constructing a vector database containing over a million documents.

These performance gains can be seamlessly integrated into RAG pipelines developed with Langchain and OpenVINO, empowering businesses to create scalable AI solutions. Such solutions can overcome many of the limitations associated with LLMs by utilizing cutting-edge information retrieval methods and the Lenovo ThinkSystem SR650 V3 with the AI acceleration features of 5th Gen Intel Xeon Scalable processors. The RAG pipeline's effectiveness can be further improved by incorporating additional components for query rewriting and document summarization.

For More Information

For more information, see the following resources:

- Artificial Intelligence solutions from Lenovo: https://www.lenovo.com/ai
- ThinkSystem SR650 V3 datasheet: https://lenovopress.lenovo.com/datasheet/ds0143-lenovo-thinksystem-sr650-v3
- Intel AI Performance: https://www.intel.com/content/www/us/en/now/ai-performance.html

Authors

Rodrigo Escobar, Ph. D. is a Systems and Cloud Solutions Engineer and AI Systems Performance Tech Lead in Intel's Data Center and AI (DCAI) organization where he works with industry partners to fine-tune workloads and improve overall system performance. His interests are in the area of industry standard benchmarks, performance analysis and optimization of AI and large-scale data analytics applications. Rodrigo is an active member of the Transaction Processing Performance Council (TPC) and Chairman of the data analytics TPCx-BB benchmark.

Abirami Prabhakaran is a Principal Engineer in Intel's Data Center and AI (DCAI) organization. Part of the Market Readiness team, she is the solution architect for end-to-end AI solutions. Her focus includes enablement and performance optimizations of AI and analytics use cases, distributed infrastructure performance and power optimization.

David Ellison is the Chief Data Scientist for Lenovo ISG. Through Lenovo's US and European AI Discover Centers, he leads a team that uses cutting-edge AI techniques to deliver solutions for external customers while internally supporting the overall AI strategy for the World Wide Infrastructure Solutions Group. Before joining Lenovo, he ran an international scientific analysis and equipment company and worked as a Data Scientist for the US Postal Service. Previous to that, he received a PhD in Biomedical Engineering from Johns Hopkins University. He has numerous publications in top tier journals including two in the Proceedings of the National Academy of the Sciences.

Ajay Dholakia is a Principal Engineer, Master Inventor, AI Leader and Chief Technologist for Software & Solutions Development with Lenovo ISG. His current focus is on solution architectures in the areas of AI / ML, Generative AI, Data Analytics, Edge Computing, and Blockchain. In his more than 30 years with Lenovo, and IBM before that, Ajay has led diverse projects ranging from research and technology to product development, as well as business and technical strategy. Ajay holds more than 60 patents and has authored over 60 technical publications including a book. He received PhD in Electrical and Computer Engineering from N.C. State University and MBA from Henley Business School.

Mishali Naik, Ph.D. is a Sr. Principal Engineer in Intel's Data Center and AI (DCAI) organization. She is currently leading Enterprise Solutions Engineering for the Market Readiness organization. Her interests include AI, computer systems and distributed architecture, application-level performance analysis and optimization, integrated HW-SW solutions and co-design, as well domain-specific customization.

Related product families

Product families related to this document are the following:

- Artificial Intelligence
- ThinkSystem SR650 V3 Server

Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area. Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service. Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.
8001 Development Drive
Morrisville, NC 27560
U.S.A.
Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION ”AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary. Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk. Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

This document, LP2025, was created or updated on September 16, 2024.

Send us your comments in one of the following ways:

- Use the online Contact us review form found at:
  https://lenovopress.lenovo.com/LP2025
- Send your comments in an e-mail to:
  comments@lenovopress.com

This document is available online at https://lenovopress.lenovo.com/LP2025.

Trademarks

Lenovo and the Lenovo logo are trademarks or registered trademarks of Lenovo in the United States, other countries, or both. A current list of Lenovo trademarks is available on the Web at https://www.lenovo.com/us/en/legal/copytrade/.

The following terms are trademarks of Lenovo in the United States, other countries, or both:
Lenovo®
Neptune®
ThinkSystem®
XClarity®

The following terms are trademarks of other companies:

Intel®, OpenVINO®, and Xeon® are trademarks of Intel Corporation or its subsidiaries.

Linux® is the trademark of Linus Torvalds in the U.S. and other countries.

Other company, product, or service names may be trademarks or service marks of others.