PURESTORAGE®
Uncomplicate Data Storage, Forever

BUSINESS WHITE PAPER

Pure Storage GenAI RAG with NVIDIA

Contents

Executive Summary

In the race to increase customer engagement, many organizations are turning to artificial intelligence (AI) to boost operational efficiencies and gain competitive advantages. However, organizations can face challenges when they lack their own data to train models, resulting in outdated answers and hallucinations that can slow enterprises down with inaccurate results—or stop them in their tracks. This retrieval augmented generation (RAG) white paper from Pure Storage with NVIDIA shows how enterprises can utilize large language models (LLMs) with vector databases to jumpstart their generative AI projects by augmenting them with corporate, external, or proprietary data to extract insights with custom responses that are more relevant, precise, and accurate. A conversational-style user interface delivers an intuitive, easy-to-use, real-time, and cost-efficient assistant for a range of enterprise tasks. AI applications leveraging this generative AI LLM RAG white paper include use cases like answering questions, summarizing documents, generating relevant marketing content, and assisting with writing code.

Introduction

Using a RAG approach over querying standard foundational LLMs has many advantages, including greater response accuracy and safety with fewer hallucinations, fewer vague, unusable, and detrimental responses. Additionally, running RAG and foundational LLMs on the user's infrastructure avoids transferring sensitive data outside the organization as when using a cloud-hosted LLM service. Vector databases are a crucial component of RAG models, enabling the efficient storage, indexing, and retrieval of additional data that could be proprietary and/or external to data found in standard LLMs. These databases allow high-dimensional vector embeddings that are used to augment the generation process. Special features of vector databases like similarity search and scalability make them well-suited for powering the retrieval step in RAG.

Benefits of Running Vector Databases on FlashBlade

Vector databases differ from traditional databases in that they are designed specifically to handle vector embeddings as numerical arrays that represent the characteristics of an object. This specialized data model allows for scalability and efficient vector data storage and retrieval,making the database more flexible. When doing high-speed search, performance is critical in vector databases, as they leverage advanced indexing techniques like locality-sensitive hashing (LSH) and approximate nearest neighbor (ANN) algorithms to enable fast similarity searches. Pure Storage® FlashBlade® systems provide native multiprotocol access for NFS, S3, and SMB and can support billions of files and objects in a single system, making them well suited for storing RAG vector databases.

Pure Storage FlashBlade//S™ is the only scale-out storage platform that efficiently powers your modern unstructured data needs, delivering cutting edge capabilities without complexity. It allows capacity and performance to be scaled independently and can be continuously improved over time. This unique modular storage architecture offers flexible consumption choices while minimizing waste by using less power and space.

When the Milvus vector database is included into our architecture, you get:

- Performance gains from Pure Storage FlashBlade//S, a unified fast file and object storage platform, even when compared to direct attached storage. This means that users will not be slowed down by storage hardware that can not keep up with the GPUs.
- Industry-leading performance density, energy, and cost efficiency. With our storage densities today, and the lower capacity of COTS SSDs that our competitors must deploy to hit the same performance and capacity targets, we routinely take up 80% less rack space than our competitors. Additionally, our efficiency advantages translate to a 2-5x reduction in energy and floor space consumption compared to competitive all-flash systems, and they offer an up to 10x reduction compared to all-HDD systems[1]. For many users, the data center real estate, energy requirements, and TCO of power-hungry AI workloads are a major concern when accommodating these workloads in existing or planned data center footprints and budgets, and Pure's architecture is designed to reduce the power consumption and TCO.
- The ability to scale VectorDBs into the 100's of terabytes and the entire data needs for RAG to 10's of PBs, while independently scaling to multiple compute GPUs. By scaling performance and capacity independently, you save money by only scaling what you need, when you need it.
- Operational simplicity with storage infrastructure that can upgrade non-disruptively with no performance reduction. This greatly enhances the user experience and productivity of our customers for growing RAG use cases.

Our RAG architecture is designed to include the Milvus vector database—a critical component of the RAG platform—which is hosted on a Flashblade//S 500. This combination of hardware and software enables a 36% performance gain when ingesting vectors compared to the default solution using local SSDs. Flashblade//S500 provides scalability advantages by providing an easy way to grow the vector database from gigabytes to hundreds of terabytes or even petabytes—something that is very challenging and resource-intensive using local SSDs.

Solution Benefits

The RAG platform from Pure Storage, with NVIDIA, outlined below delivers unparalleled performance, scalability, and ease of use compared to the competition. This innovative solution leverages the power of vector databases and advanced indexing to revolutionize the way organizations access and utilize their valuable data. **The platform includes:**

- **Seamless data integration**: The RAG platform seamlessly integrates a wide range of documents, including brochures, internal web pages, emails, videos, photos, knowledge base articles, and more. These assets are converted into vectors that are embedded into a vector database, and an index is built to establish a direct link between the documents and the user's queries.
- **Intelligent query matching**: The user's queries are also vectorized using the same embedding model, allowing for precise and similarity search matching against the vector database. This process ensures that the most relevant documents are retrieved, providing users with the information they need to enhance their queries and the overall context.
- **Augmented query processing**: The retrieved documents are then used to augment the user's query and provide additional context to the LLM. This integration of relevant information helps to improve the accuracy, timeliness and relevance of the LLM's responses, ensuring that users receive the most accurate and valuable information.
- **Robust guardrail filtering**: The retrieved documents can also be utilized to fine-tune the guardrails that filter out bad inbound queries and/or inappropriate outbound responses. This feature helps to maintain the integrity and safety of the system, ensuring that users receive only the most appropriate and trustworthy information, as well as reducing the risk of using the LLM maliciously.

- **Unparalleled performance and scalability**: The Pure Storage RAG platform with NVIDIA delivers unparalleled performance and scalability, allowing organizations to efficiently manage and access their vast data repositories. This solution is designed to handle even the most demanding data requirements, ensuring that users can access the information they need quickly and seamlessly.
- **Ease of use and integration**: The RAG platform is designed with user-friendliness in mind, making it easy to integrate into existing workflows and systems. This streamlined approach allows organizations to leverage the power of this innovative solution without the need for extensive technical expertise or resources.

Solution Use Cases

RAG introduces a transformative approach for integrating large language models with an organization's specific knowledge and data. This synergy between the deep learning capabilities of an LLM and an organization's unique data sets enables RAG to address the challenges associated with deploying LLMs for real-world applications effectively. Below are detailed explanations of how RAG benefits enterprises across various use cases.

Semantic Search and Efficient Information Retrieval

RAG significantly enhances the capabilities of semantic search, making it a powerful tool for enterprises to navigate and extract pertinent information from their extensive data repositories. This is particularly beneficial in scenarios where precision and context are paramount. For example, a legal firm could leverage RAG to pinpoint relevant case precedents quickly, or a pharmaceutical company might use it to efficiently locate specific drug research papers. By understanding the intent behind queries and matching them with contextually relevant content, RAG streamlines the process of information retrieval, making it more efficient and accurate.

Enhancing Customer Service

In the realm of customer service, RAG offers substantial improvements in both efficiency and the quality of customer interactions. Traditional customer service chatbots often provide responses based on a limited set of pre-defined answers, which may not always address the customer's specific needs accurately; and the predefined answers are not updated often. However, with RAG, a chatbot can access and understand an organization's unique policies, customer details, and proprietary information–including information discovered or generated earlier the same day—enabling it to deliver precise and personalized responses to customer inquiries. This capability is not limited to one specific industry; it extends to any customer service scenario where personalized, accurate information delivery is crucial.

Improving Content Creation

RAG also plays a pivotal role in content creation, aiding creators in producing high-quality, context-aware content. This is achieved through several mechanisms:

- **Summarizing articles:** RAG can efficiently condense lengthy documents into concise summaries, preserving the core message and relevant details. This is particularly useful for content creators who need to quickly grasp the essence of extensive materials.
- **Recommendations:** By analyzing retrieved information, RAG can suggest related content, enabling creators to explore and integrate complementary topics or ideas into their work. This feature enhances the depth and relevance of the content being produced.
- **Generating new pieces**: RAG's ability to generate text based content on retrieved data opens up possibilities for creating personalized marketing emails, blog posts, and other forms of content. By tapping into an organization's data, RAG can produce content that is not only more relevant but also closely tailored to the specific audience or purpose.

Solution Design

The importance of a RAG pipeline lies in its ability to provide LLMs with relevant and factual information to reduce the likelihood of hallucinations, which are faulty but convincing responses when LLMs are not supplied with accurate data. RAG also allows for dynamic data integration, ensuring the system's responses are based on the latest available information. Additionally, RAG can enhance the transparency and trust of AI systems by enabling users to trace back how the AI came to a response, as it fetches and presents data from specific, verifiable sources. A deeper look at the architecture design from Pure Storage with NVIDIA will give a deeper understanding of the solution.

The RAG pipeline consists of the following stages:

- **Finding relevant data sources**: Identify data sources that contain relevant information for your task. New data can be added as often as daily.
- **Getting raw data**: Retrieve raw data stored in various formats such as text, PDF, or images.
- **Document preprocessing/chunking**: Parse the raw data into smaller "chunks" (sentences or paragraphs) to help the model understand the essence of the content. This step prevents important facts from being lost during further processing.
- **Generating embeddings**: Create high-dimensional vector representations (embeddings) for the processed data. These embeddings retain the semantics of the text.
- **Storing embeddings in the VectorDB**: Process the data to make it compatible with the VectorDB's API (commonly using JSON , Parquet or NumPy files). This approach can unlock up to 20% more capacity.
  - Transfer files containing the vector embedding of the data to the FlashBlade objectstore if the VectorDB API only supports accessing data from objectstore.
  - Insert the vector embeddings to a VectorDB (VectorDB may store the files on FlashBlade objectstore or NFS filesystem).
  - Optionally, perform embedding during this phase for more control over the model selection.
  - FlashBlade DirectFlash Modules manage storage on a global level, utilizing NVRAM for scalability.
- **Indexing data**: Vector indexes are an organizational unit of metadata used to accelerate vector similarity search.
  - Build the index by specifying the vector field name and index parameters which specify the index type to use.
- **Retrieval:** Before querying the LLM, compare your query against the vector DB, which contains millions or billions of vectors.
  - Retrieve the most semantically similar documents and associated metadata.
  - Low latency is crucial, but the quality of retrieval depends on the combination of embedding and latency—fast retrieval is useless if the results are irrelevant or low-quality.
  - Reranking: Optionally, the retrieved content can be reranked for greater relevance to the query
- **Generation:** The last generation step in the RAG pipeline combines the user's query and the retrieved context from the vector DBtransforms and transforms it into meaningful knowledge by generating contextually rich and accurate responses. RAG answers combine the knowledge of the pre-trained LLM with the most relevant content retrieved from the vector DB.

The solution design includes the following components:

- **NVIDIA NIM inference microservices**: NVIDIA NIM is a collection of easy-to-use microservices for accelerating the deployment of foundation models. NIM is designed to bridge the gap between the complex world of AI development and the operational needs of enterprise environments, enabling 10-100X more enterprise application developers to contribute to AI transformations of their companies.
- **NVIDIA NeMo Retriever**: NVIDIA NeMo Retriever is a collection of microservices enabling semantic search of enterprise data to deliver highly accurate responses using retrieval augmentation. With NeMo Retriever, enterprises can simplify and accelerate the document embedding, retrieval, and querying functions at the heart of a RAG pipeline. This architecture uses the NeMo Retriever microservices to accelerate GPU-based vector embedding computations.
- **NVIDIA API catalog**: In the NVIDIA API catalog, developers can explore the latest community-built AI models with APIs optimized and accelerated by NVIDIA, then deploy them anywhere with NVIDIA NIM.

Prerequisites:

Component	Functionality
NVIDIA AI Enterprise	End-to-end AI platform accelerates data science pipelines and streamlines development and deployment of production-grade co-pilots and other generative AI applications.
NVIDIA NIM	NVIDIA NIM, part of NVIDIA AI Enterprise, is a set of easy-to-use microservices designed to speed up generative AI deployment in enterprises. Supporting a wide range of AI models, including NVIDIA AI foundation and custom models, it ensures seamless, scalable AI inferencing, on-premises or in the cloud, leveraging industry standard APIs.
NVIDIA NeMo Retriever	NVIDIA NeMo Retriever, part of NVIDIA AI Enterprise, provides world-class information retrieval with the lowest latency, highest throughput, and maximum data privacy, enabling organizations to make better use of their data and generate business insights in real-time. NeMo Retriever enhances generative AI applications with enterprise-grade retrieval-augmented generation (RAG) capabilities which can be connected to business data wherever it resides.
NVIDIA GPU Operator	Part of NVIDIA AI Enterprise, the NVIDIA GPU Operator uses the operator framework within Kubernetes to automate the management of all NVIDIA software components needed to provision GPUs.
Kubernetes	Container orchestration platform
OEM Servers Equipped with NVIDIA A100 GPUs	Accelerated compute server
NVIDIA BlueField-3 DPU	Accelerated Server Networking
FlashBlade//S 500, 1 - 3 chassis	Scale-out Object and File Storage
NVIDIA Spectrum SN3700 Ethernet Switch	Network Fabric

Recommended Hardware Architecture and Design

For the testing and design of the Pure Storage and NVIDIA integrated solution, we used 2 GPU nodes with 1 3-chassis FlashBlade//S 500, and NVIDIA Spectrum-2 SN3700 Ethernet switches.

- **Server**: NVIDIA-certified 2 GPU Nodes. OEM servers equipped with NVIDIA A100, L40S, or H100 Tensor Core GPUs are powerful compute systems for scale-out AI infrastructure and workloads, ranging from analytics to training and inference. Integrating NVIDIA BlueField-3 DPUs in every compute server accelerates GPU access to networked storage, supercharging application performance.
- **Storage**: 1 3-chassis FlashBlade//S 500. FlashBlade//S 500 excels at energy efficiency, scalability, multi-modal performance and is designed to handle structured and unstructured data efficiently. The modular architecture of FlashBlade//S allows you to independently scale capacity and performance, which ensures greater efficiency and minimizes waste.
- **Network switch**: NVIDIA Spectrum-2 SN3700. The Spectrum-2 SN3700 enables connectivity to endpoints at different speeds and carries a throughput of 6.4Tb/s, with a landmark 8.33Bpps processing capacity. As an ideal spine solution, the SN3700 allows maximum flexibility, with port speeds spanning from 10GbE to 200GbE per port.[1]

Conclusion

RAG represents a significant advancement in the application of LLMs within enterprises. By bridging the gap between the generalized capabilities of LLMs and an organization's specific, proprietary knowledge, RAG enables more effective and efficient use of AI in semantic search, customer service, and content creation. Its ability to understand and retrieve contextually relevant information from vast data repositories makes it an invaluable tool for enterprises looking to leverage AI for competitive advantage.

The Pure Storage RAG platform with NVIDIA represents a significant advancement in data retrieval and LLM augmentation, offering organizations a powerful and versatile solution that delivers unparalleled performance, efficiency, and ease of use. By combining the expertise of these industry leaders, this platform is poised to transform the way organizations access and utilize their valuable data.

Pure Storage and NVIDIA are collaborating on a set of reference architectures for various generative AI and RAG use cases, giving our customers a frictionless way to use the power of their data with RAG solutions for different verticals which are scalable, performant and efficient. This is the start of a series of technical documents on scalable production optimized RAG architectures from Pure Storage.

Additional Resources

- Learn how you can accelerate adoption of AI with the Pure Data Storage Platform.
- Explore AI Ready Infrastructure that simplifies enterprise AI.
- Discover FlashBlade//S for all of your unstructured data storage needs.
- Explore Github for the FB py-client, and remember that the REST API is built in.

1 https://www.purestorage.com/resources/efficient-it-infrastructure-saves-more-than-just-energy-costs.html

PS2611-01-en 06/24