Microsoft

Enhancing efficiency in AVOps with Generative AI

Authors:

Ryan Matsumura – Senior Program Manager

Jochen Schroeer – Senior Lead Architect Mobility

Introduction

Did you know that human error is responsible for more than 90% of road crashes in the United States[1]? According to the National Highway Traffic Safety Administration (NHTSA), automated driving systems (ADS) and advanced driver assistance systems (ADAS) have the potential to prevent or mitigate many of these crashes by reducing human errors[2].

However, developing and testing ADS and ADAS is not an easy task. Developing and testing autonomous driving systems is not only complex and costly, but also time-consuming. According to McKinsey, the additional hardware and software costs per vehicle for Level-3 and Level-4 systems could reach $5,000 or more during the early rollout phase, with development and validation costs exceeding more than $1 billion[3].

Moreover, the technologies required to achieve advanced levels of autonomous driving suggest a significantly longer timeline; such vehicles are five to ten years away[4]. This means that the auto industry faces a huge challenge to bring autonomous mobility to the mass market in a profitable and sustainable way.

How can generative AI help overcome this challenge? Generative AI is a branch of artificial intelligence that can create new content or data from scratch, such as text, images, music, or code. It can be used to augment or replace real-world data for autonomous mobility applications, such as simulation, validation, optimization, and personalization. In this paper, we explore how generative AI can enhance autonomous mobility by addressing its key challenges and opportunities. We present the top generative AI use cases for autonomous vehicle operations, such as generating synthetic scenarios, validating sensor data, optimizing driving behavior, and personalizing user experience. We also review the main techniques for generative AI. We discuss the current state-of-the-art and the future directions and open problems of generative AI for autonomous mobility.

Autonomous vehicle operations (AVOps)

Microsoft has created a reference architecture that helps provide a blueprint for autonomous vehicle operations (AVOps). AVOps involves the back-end of autonomous vehicle development, which includes organizations such as data operations (DataOps), ML (Machine Learning) training/operations (MLOps), and validation/verification operations (ValOps).

An organization that implements AVOps typically requires a substantial amount of storage and compute to:

Capture and process recordings of CAN (Controller Area Network) data and scenes from test vehicles as learning material for the perception models that vehicles need to drive autonomously.

Train perception models based on labeled recordings, for recognizing an environment, as the base functionality for driving autonomously.

Perform safety validation based on open-loop and closed-loop simulations. In this context, finding and labeling representative recordings for training the perception stack as well as validating this stack based on scripted scenarios utilized by simulators are manual efforts where generative AI can help.

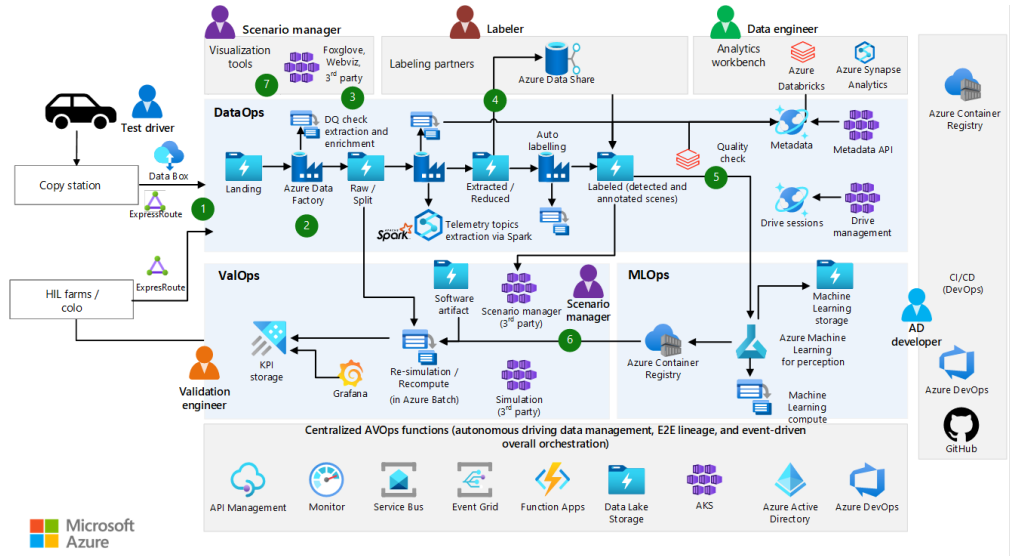

Figure 1 - AVOps Reference Architecture

Even when an organization has implemented an architecture such as that explained in Figure 1, challenges such as the following still exist such as the following:

Challenges

- As levels of autonomy increase for an automated driving vehicle, so do the number of complex and unique scenarios that the system needs to handle. Capturing real world data is impossible for all situations.
- Automated driving requires strict adherence to industry standards such as ISO 26262. An organization implementing any such solutions that require adherence to ISO 26262 can be encumbered by many manual tasks such as requirements or test case management. In addition, there might be major inconsistencies due to the manual nature of these tasks.
- The number of lines of code in a vehicle today is approximately 100 million. According to McKinsey, in a report for Cybersecurity, the expectation is that by 2023 there will be roughly 300 million lines of code in a vehicle[5]. The number of software developers required is monumental and the industry needs to rethink how software should be developed. In addition, the growth of software in vehicles also leads to an increase in security vulnerabilities that has tremendous impact on safety and cost to the OEMs (Original Equipment Manufacturers) and consumers.
- Autonomous companies and suppliers are dealing with enormous amounts of data. So, the ability to search for data requires you to perform time-consuming tasks of labeling even when the data is auto-labeled.

Potential for generative AI

Generative AI is a branch of artificial intelligence that can create novel and diverse data from existing data sources. It has the potential to enhance autonomous vehicle development and operations (AVOps) in many ways, such as the following:

- New enhancements to foundational models improve search recommendations with natural language queries that can measure the similarities between images and text.
- AI Copilots can be used to assist with various tasks such as rating, summarizing, elaborating, converting, and translating requirements and scenarios. Copilots can improve the quality, speed, and cost of software development by reducing errors and rework.
- An AI pair programmer can help write code not only faster than humans but also be more consistent with coding and safety standards for automotive requirements.
- Generating realistic and diverse scenarios for testing and validation of autonomous driving systems, reducing the need for costly and time-consuming real-world data collection and annotation.
- Enhancing the perception, planning, and decision-making capabilities of autonomous vehicles by augmenting the available sensor data with synthetic

data generated by generative AI models. This helps in improving the robustness and generalization of the systems.

- Security AI Copilots can help security teams protect their organizations from cyberthreats at machine speed and scale. A Copilot works with the help of a large language model that can understand natural language queries and generate security-specific responses.

Semantic scene search and identification

Auto-annotations today are typically a semi-automated process in the whole automotive industry. New video or image material is run through services trying to automatically detect objects like traffic signs, scene descriptions (overtaking of a vehicle etc.) in the images / frames which are required to identify relevant scenarios for training and validation of the autonomous perception stack. As a result, the entire process becomes cumbersome.

To improve the process, a data management foundation such as one detailed in DataOps reference architecture, is important. This foundation allows indexing, annotation, searching, and selection of relevant data sets used for training of computer vision models required for autonomous driving. Vector search and embeddings can help to identify the relevant scenes and scenarios that are annotated for the learning and validation downstream processes enhanced by manual reviews to focus on workloads for important scenes.

Figure 2 diagram depicts the workflow to support this scenario:

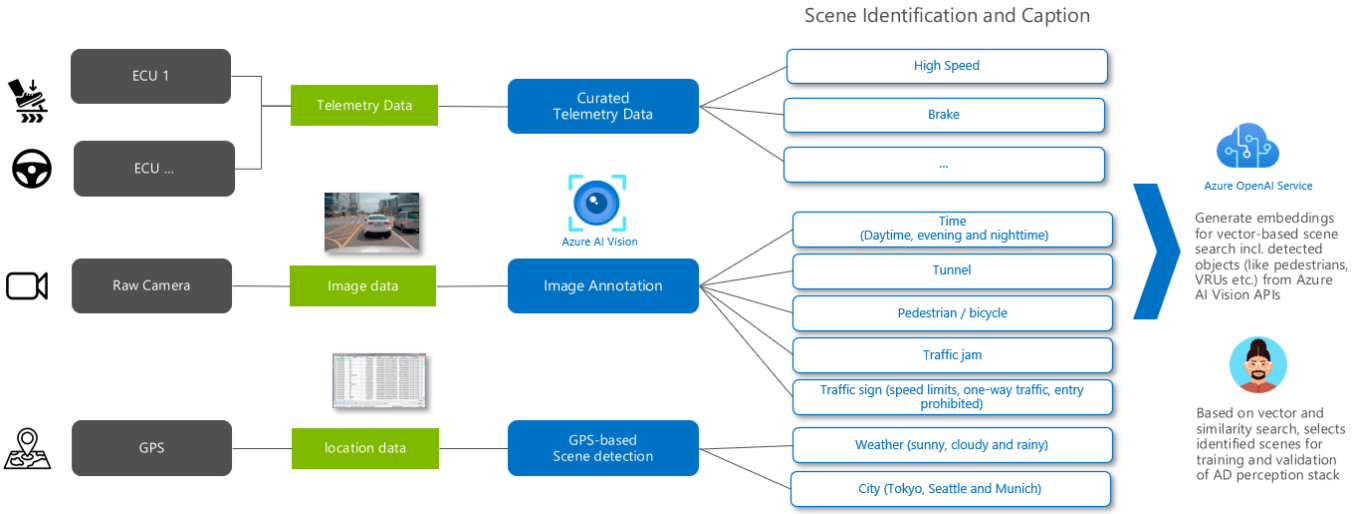

Figure 2 – Auto-Annotations and Semantic search workflow

The workflow starts with real recordings (also to derive synthetically scenarios from these at later stage based on identified scenes) from vehicles. The Azure AI Vision service auto-annotates the recordings to detect objects and gets an understanding of the recorded frames (for camera / images)

- The vehicle and specially equipped cameras already provide information such as lane change.
- Different signal streams such as camera, CAN telemetry, and locations are merged to identify and capture scenes, and get an understanding of the contained objects. As example, we partnered with TCS to enhance auto-annotation generated by Azure AI Vision service with CAN based scene detection, see [13]
- The information is passed to Azure OpenAI service to generate embedding as preparation for semantic and vector-based scene search. Generate embedding is the process of converting text into a dense vector of floating-point numbers which represent a data representation that can be easily utilized by machine learning models and algorithms.[6]

Figure 3 diagram shows the technical architecture leveraging Microsoft AI services:

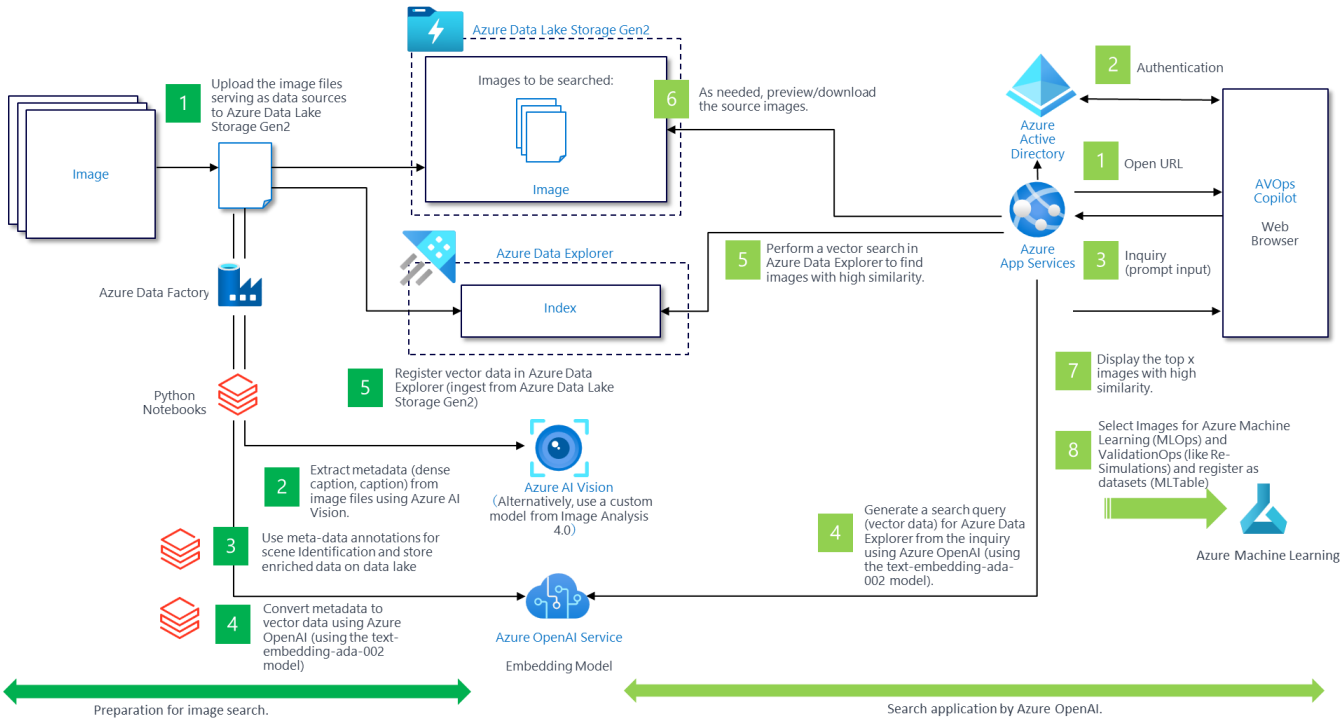

Figure 3 - Auto-Annotations and Semantic search architecture example

You can perform scene search inclusion of annotated results using, for instance, a notebook, or integrate them in custom UI web tools or Copilot for AD developers to select and extract relevant scenes:

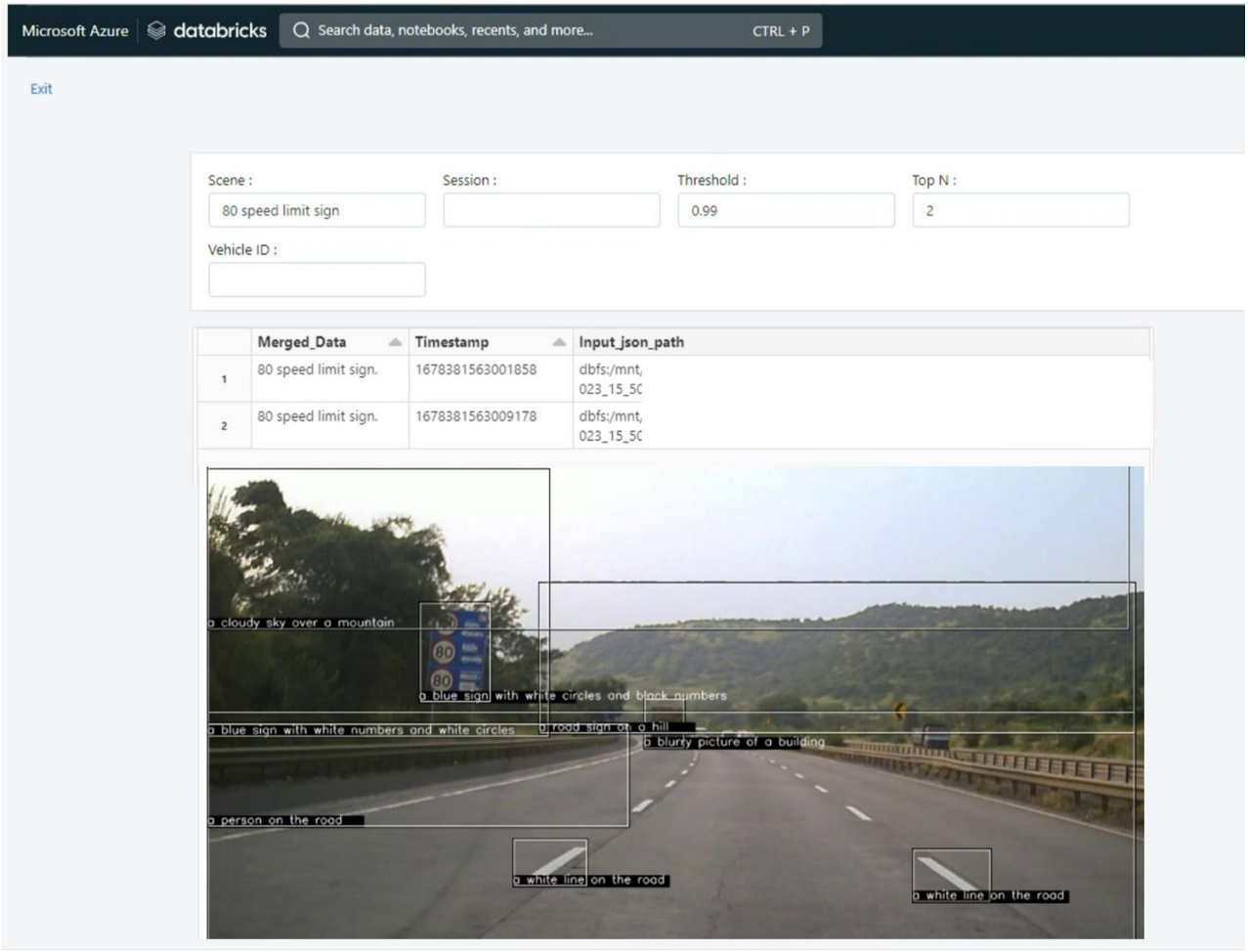

Figure 4 - Example output[13]

Generate scenarios for simulations

For autonomous driving and assistance systems, you must perform open and closed loop validations by simulating specific scenarios. OpenDRIVE and OpenSCENARIO are examples of widely adopted industry standards to define the driving scenarios. Today, data scientists often identify corner cases and then use a common scripting language such as Python to generate scenarios in a high-level script format compliant with OpenSCENARIO. Test managers manage the suite of scenarios and test engineers manage the test execution, but neither the engineers nor managers can manage the scripts.

A Copilot interface leveraging Azure OpenAI LLMs (large language models) can support the generation of OpenSCENARIO files passed to simulation engines and to make the entire validation process more efficient.

Example:

Test managers write vague scenarios and often need to iterate. A Copilot that accepts the test manager's first draft of a scenario, such as *Two cars on a highway overtaking each other,* and provides a response on refining the scenario and supporting the script generation process.

This end-to-end process can look like the following:

1. Utilize generative AI to create scenarios (fine-tuned model based on existing scenario database)
2. Based on the created scenarios, generate test scripts / open scenario files that comply with OpenSCENARIO and input them into the simulator
3. Run the simulation
4. If the test scenario fails, add similar test scenarios and variations, and refine these further. (See PEGASUS-Gesamtmethode.pdf (pegasusprojekt.de)).

Through such a process, the intention is to evaluate algorithms in more 'longtail' specialized cases. The diagram shows the architecture leveraging Microsoft AI capabilities to support this refinement process:

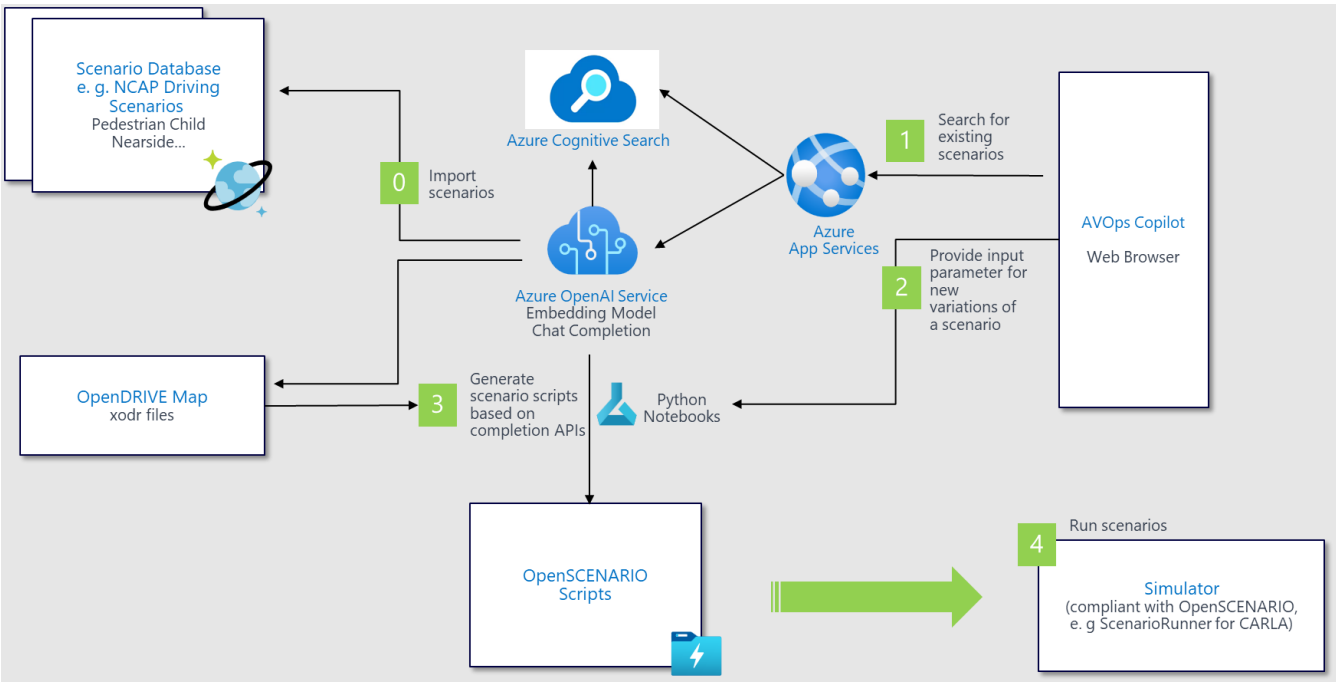

Figure 5 - Example Architecture leveraging Microsoft AI capabilities for refinement process

Generating data from real-world data

One of the challenges of autonomous driving development is to collect and label enough data to train and test the AI models that control the vehicle. However, data

collection is costly, time-consuming, and limited by the availability of real-world scenarios. Moreover, data labeling is prone to human errors and inconsistencies, and requires a lot of manual effort. Therefore, it is desirable to have a way of generating new data from existing data, without compromising the quality and diversity of the data.

NeRF (Neural Radiance Fields) is a technology that can help with this problem. NeRF, which stands for neural radiance fields, was first introduced in 2020 by a team from UC Berkeley, Google Research, and UC San Diego in a paper titled *NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis*[7]. NeRF is a way of representing and rendering realistic 3D scenes based on an input collection of 2D images. NeRFs use a neural network to learn how light rays interact with the scene, and then synthesize novel views of the scene from any angle. NeRFs can produce high-quality and photorealistic 3D models that capture minute details and textures of the scene.

Figure 6 - Image Source: NeRF Paper (Mildenhall1, Srinivasan1, Tancik, et. al), 2020

NeRF can help with autonomous driving development by creating new data from existing data in two ways:

Data augmentation: NeRF can use a collection of 2D images captured by cameras mounted on a real vehicle to reconstruct a 3D scene of the environment. Then, NeRF can generate new images of the scene from different viewpoints, angles, and distances, which can be used to augment the original data set. This can increase the amount and variety of data available for training and testing the AI models and improve their generalization and robustness for corner cases.

Data simulation: NeRF can also use a collection of 2D images captured by cameras mounted on a real vehicle to reconstruct a 3D scene of the environment. Then, NeRF can separate moving objects from static environments, change the original trajectories of objects, add new objects and trajectories, and add uncommon traffic participants. This can create diverse and realistic simulation scenarios that can be used to train and test the AI models in different situations and conditions.

NeRF is a powerful tool that can help with autonomous driving development by generating new data from existing data. It can reduce the cost and effort of data collection and labeling and increase the quality and diversity of data for training and testing the AI models.

One of Microsoft's key partners, NVIDIA, uses NeRF in their product DRIVE Sim as part of their Neural Reconstruction Engine.[8] DRIVE SIM today runs on Omniverse on Azure. In addition, WAYVE uses NeRF to automatically generate photorealistic 3D worlds and scenarios from real-world driving data using Azure Cloud capabilities.[9]

Requirements Management

Managing requirements in the automotive industry entails unique challenges, demanding precision, timing, and adherence to strict safety standards such as ISO 26262. Generative AI can help with developing autonomous vehicles by generating and managing complex requirements that involve safety, functionality, performance, and user experience. These requirements must comply with the standards of ASPICE and ISO 26262, the current benchmarks for software quality and safety in the automotive industry. An AI Copilot for requirements management can assist with various tasks such as rating, summarizing, elaborating, converting, and translating requirements. It can improve the quality, speed, and cost of software development by reducing errors and rework.

A key challenge with using generative AI is the data that a model like GPT4 is trained on. This is especially of concern when considering the vast amounts of requirements, documents, and internal standards that GPT4 is not privy to. A key design pattern that can be used to augment the capabilities of an LLM (large language model) like ChatGPT is Retrieval Augmentation Generation (RAG). RAG architecture means that you can constrain natural language processing to your enterprise content sourced from vectorized documents, images, audio, and video.

Combining cognitive search with LLM allows organizations to provide inputs to the LLM prompt but does not train the model. In RAG architecture, there is no extra training. The LLM is pretrained using public data, but it generates responses that are augmented by information from the retriever.

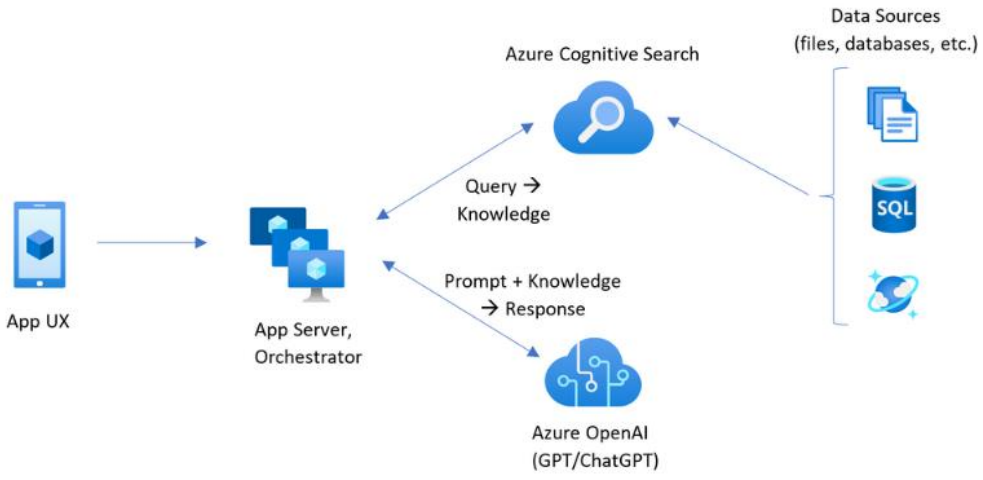

Figure 7- RAG Pattern including Cognitive Search

For more information, see RAG and generative AI - Azure Cognitive Search | Microsoft Learn and Azure/GPT-RAG GitHub Reference.

The Azure OpenAI *Use your data* feature (preview) makes this process for RAG even more efficient. For more information, see Use your own data with Azure OpenAI Service - Azure OpenAI | Microsoft Learn.

Microsoft partners such Modern Requirements have built a solution that can help customers called Copilot4DevOps. With the use of Azure DevOps, Azure OpenAI, and Cognitive Services, Modern Requirements Copilot4DevOps significantly enhances the productivity and efficiency of development teams.

Copilot4DevOps is an AI-powered tool that helps teams automate a large portion of the work, it allows teams to concentrate on strategic decision-making. This leads to faster authoring of requirements, improved analysis, and a reduction in errors.

In addition to individual productivity, Copilot4DevOps offers several organizational benefits. It enhances the quality of documentation, accelerates the time to market, and reduces the overall cost of projects. The AI assistance provided by Copilot4DevOps is a key feature that increases employee satisfaction and productivity. It also prioritizes security, offering the latest updates and protocols from Microsoft and OpenAI. If needed, the AI integration can be turned off at the admin level for privacy and security reasons.

With Copilot4DevOps, you can generate high-quality requirements from raw data with just one click. It allows you to analyze work item data for clarity, conciseness, coherence, correctness, courtesy, and conviction. It also helps in creating comprehensive use cases or user stories from the actor and user perspective, enhancing project clarity.

Finally, Copilot4DevOps offers the ability to convert your requirements data into Gherkin format, a language used for writing software behavior specifications. Overall, Copilot4DevOps provides a secure and controlled environment for managing your development requirements. It's a powerful tool that can greatly enhance your development process.

AI pair programmer

Generative AI can help with autonomous vehicle development by providing solutions for various challenges, such as enhancing developer productivity and creativity by reducing boilerplate, errors, and rework. Developer tools must support the best practices and processes recommended by Automotive SPICE (ASPICE)[10], which covers how to conduct software and systems design. Developer tools must also integrate with popular editors and frameworks to provide real-time suggestions and feedback.

One example of a generative AI tool that can help with autonomous vehicle development is GitHub Copilot, an AI pair programmer that helps developers write

code faster and with less work. GitHub Copilot is powered by OpenAI which uses a large language model like GPT-3.5 Turbo that translates natural language into code. GitHub Copilot can draw context from comments and code to suggest individual lines and whole functions across dozens of languages.

GitHub Copilot can assist developers with various tasks, such as:

- Writing clear and understandable comments to describe the logic and functionality of the code.
- Generating code snippets or functions based on natural language prompts or existing code.
- Reducing boilerplate, errors, and rework by filtering out common vulnerable coding patterns and suggesting best practices.
- Speeding up test generation by creating test cases or scenarios based on the code or specifications.

GitHub Copilot can also help developers learn new languages or frameworks, or explore new domains or technologies, by providing relevant examples and documentation. GitHub Copilot can improve developer productivity, satisfaction, and innovation by allowing them to focus on solving bigger problems and creating more value.

Experiment results conducted by the GitHub team showed that developers who used GitHub Copilot completed a coding task significantly faster than those who did not use it. The developers using GitHub Copilot also had a higher rate of completing the task successfully. The results show that developers who use GitHub Copilot complete the task significantly faster (55%) and more successfully (78%) than those who do not. For more information, see the referenced blog post.[11]

Generative AI is a powerful and promising technology that can help with autonomous vehicle development by providing solutions for various challenges, such as generating and managing complex requirements that comply with ASPICE and ISO 26262 standards, testing and validating software and hardware components that meet functional safety goals and criteria, and enhancing developer productivity and creativity by reducing boilerplate, errors, and rework.

Security Copilot

A security copilot is an AI-powered security solution that helps security teams protect their organizations from cyber threats at machine speed and scale. It works by using a large language model that can understand natural language queries and generate security-specific responses. It also leverages the company's security expertise and global threat intelligence to provide tailored insights and recommendations.

Security copilot can help security teams with various tasks, such as:

- Writing complex queries based on natural language questions.

- Summarizing security incidents and providing actionable guidance.
- Creating effective and efficient test cases for different scenarios and datasets.
- Analyzing signals and data from multiple sources and identifying potential gaps or errors.
- Optimizing the system requirements and design alternatives for autonomous vehicles.
- Accessing and integrating companies finished threat intelligence.

An example of such a security copilot is Microsoft's Security Copilot. Security Copilot is an AI assistant for security teams that builds on the latest large language models and harnesses Microsoft security expertise and global threat intelligence to help security teams. Microsoft receives over 65 trillion signals a day. Security Copilot is already helping preview customers save up to 40 percent of their time on core security operations tasks with capabilities such as writing complex queries based only on natural language questions and summarizing security incidents.[12]

Security Copilot is embedded within Microsoft 365 Defender, a leading extended detection and response (XDR) platform, to provide a unified and seamless experience for security analysts. Security Copilot also includes Microsoft Defender Threat Intelligence at no cost, which enables customers to directly access and operate on Microsoft's threat intelligence.

Security Copilot is designed to be a trusted and responsible AI solution that protects the data and privacy of customers. It does not use the data provided by customers to train other AI models, and it follows the industry-leading compliance and security controls.

Summary

In this paper, we have explored how generative AI, a branch of artificial intelligence that can create new content or data from scratch, can enhance autonomous mobility, and the ability of vehicles to operate without human intervention. We have discussed the key challenges and opportunities of generative AI for autonomous vehicle operations (AVOps), such as simulation, validation, optimization, and personalization. We have presented the top generative AI use cases for AVOps, such as generating synthetic scenarios, validating sensor data, optimizing driving behavior, and personalizing user experience. We have discussed the current state-of-the-art and the future directions and open problems of generative AI for autonomous mobility.

We conclude that generative AI can help reduce the complexity, cost, and time of developing and testing autonomous driving systems, and improve their safety, efficiency, and user satisfaction. Using Microsoft's AVOps reference architecture and Microsoft AI services, we believe that we can help solve significant challenges that the industry faces.

[1] https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety
[2] https://www.nhtsa.gov/vehicle-manufacturers/automated-driving-systems
[3] https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/autonomous-drivings-future-convenient-and-connected
[4] https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/self-driving-car-technology-when-will-the-robots-hit-the-road
[5]https://www.mckinsey.com/~/media/mckinsey/industries/automotive%20and%20assembly/our%20insights/cybersecurity%20in%20automotive%20mastering%20the%20challenge/cybersecurity-in-automotive-mastering-the-challenge.pdf
[6] https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/embeddings
[7]https://datagen.tech/guides/synthetic-data/neural-radiance-field-nerf/
[8] https://blogs.nvidia.com/blog/2021/11/09/drive-sim-replicator-synthetic-data-generation/
[9] https://wayve.ai/neural-rendering/
[10] https://vda-qmc.de/en/automotive-spice/
[11] https://github.blog/2022-09-07-research-quantifying-github-copilots-impact-on-developer-productivity-and-happiness/
[12] https://www.microsoft.com/en-us/security/blog/2023/10/19/microsoft-security-copilot-early-access-program-harnessing-generative-ai-to-empower-security-teams/
[13] Bringing Life to Smart and Sustainable vehicles with TCS