AECC

AUTOMOTIVE EDGE
COMPUTING CONSORTIUM

AECC PROOF OF CONCEPT

AI Agents Utilizing End-to-End LLMs

By Toyota, Oracle, and KDDI

Version 1.1

Abstract

Voice-activated AI has the potential to make driving safer and more efficient and to elevate vehicles to the status of companions. When given the choice, people want to interact with their cars in almost the same way that they would talk with another person. Specifically, we want AIs to factor in varied situational contexts (including the external environment and subjective considerations like our tone of voice), conversation history, and even demonstrate the ability to respond based on inference.

AECC's latest proof of concept (PoC) takes an important step towards making this idea a reality. In these demonstrations, our engineering team used voice user interfaces (VUIs) based on large language models (LLMs), which are highly advanced AIs trained on massive natural language datasets. However, relying on a single, centrally located LLM doesn't provide the kind of high-quality and diverse interaction that consumers want, and because they're centrally located, they don't provide the same experience to drivers in all locations.

For the most seamless and natural experience, our team used multiple LLMs within our voice-activated system. This was integrated within an end-to-end platform that spans in-vehicle devices, edge data centers, and public cloud services.

The PoC aimed to enable drivers to engage in natural conversations by selecting the most appropriate AI agent for the type of interaction they desired. The AI agents were deployed in the best environment based on their characteristics, either the in-vehicle device, an edge data center, or the cloud. This system enabled the execution of multiple scenarios that were previously too complex for traditional VUIs to handle.

The results confirmed that AECC technologies are ideal for deploying AI agents effectively so they can contribute to a better driving experience.

Like all AECC PoCs, this prototype system could not have happened without cooperative effort between AECC member companies. Toyota contributed overall PoC planning, demo system architecture design, software prototyping, and evaluation. KDDI and Oracle contributed to the in-depth exploration of PoC concepts and extraction of future challenges.

Business Strategy

AI is becoming an important differentiator in the hyper-competitive automotive market because VUI contributes significantly to the drivers' in-vehicle experience. According to Voicebot.ai's 2020 "In-car Voice Assistant Consumer Adoption Report."[1]

- 60% of consumers say voice assistants influence their new car purchase criteria.
- 50% of consumers believe that VUI performance has not significantly improved in the past two years.

The problem with traditional voice assistants is that they only support a few predetermined commands for a few basic functions, which the driver must remember verbatim. For example, "activate dynamic radar cruise control" might work to turn on the radar cruise control, but "drive automatically" might not. Combining information from multiple sources and a flexible understanding of the driver's requests are simply not possible for simple voice assistants.

[1] https://voicebot.ai/wp-content/uploads/2020/02/in_car_voice_assistant_consumer_adoption_report_2020_voicebot.pdf

With LLM-based AIs, however, a whole new set of complex interactions becomes possible for the driver. These include:

- Small talk with the AI, for example, discussing the driver's work schedule for the day or if the driver is likely to enjoy a popular new movie or TV show.
- Guidance for tourists, including sightseeing recommendations based on personal interests and answering questions about specific landmarks detected by the vehicle's camera system (for example, "What's that building on the right?).
- Wayfinding alerts, for example, a warning about an accident that happened moments ago and a recommendation for a detour route.
- A heads up that the driver's favorite BEV charging station has available kiosks, and it would be a good opportunity to top up the vehicle's battery power.
- A request to book a service appointment for the car due to an issue with the car's systems.

Creating an AI agent of sufficient complexity and effectiveness could be the "iPhone moment" for vehicles that wins unprecedented brand loyalty and market share.

Proof of Concept Objective

This PoC aims to create and demonstrate a voice-activated AI system for connected vehicles that seamlessly integrates multiple LLMs, offering greater flexibility and functionality than traditional systems. Drivers will interact naturally with AI agents optimized for different environments — in-vehicle, at the network edge, or in the cloud.

The system's ability to handle complex conversations and tasks highlights its potential for smarter, more responsive vehicle interactions. Its flexible deployment is expected to reduce latency, boost efficiency, and enable integration with cloud-based services, enhancing overall user satisfaction.

Proof of Concept Scenario

This proof of concept involved three scenarios, in which three different LLMs and other supporting systems (including databases and application-program interfaces, or APIs) were accessed on an in-vehicle device, on a network edge server, or in the cloud. The application logic for each AI agent (an AI-powered software that performs a complex task) is implemented on the agent server.

The essential workflow in all three cases can be simplified to the following:

1. The driver asks a question or makes a request. For these scenarios, we had the driver choose which AI agent to use based on the type of query and the information resources the AI agent had access to.
2. An automatic speech recognition (ASR) tool converts the driver's speech to text so that it can be processed by the AI agent.
3. The agent server transfers all inputs to the LLM and/or systems set up for the agent. (In some cases, assets like vehicle camera footage were required, and this was part of the input package.)
4. The inputs would all be processed by the LLMs and transferred back to the agent server, which would then provide information to the AI agent.
5. The AI agent would provide a spoken answer to the driver.

For the AI agent client and server, the team used LangChain, an open source LLM application development framework.

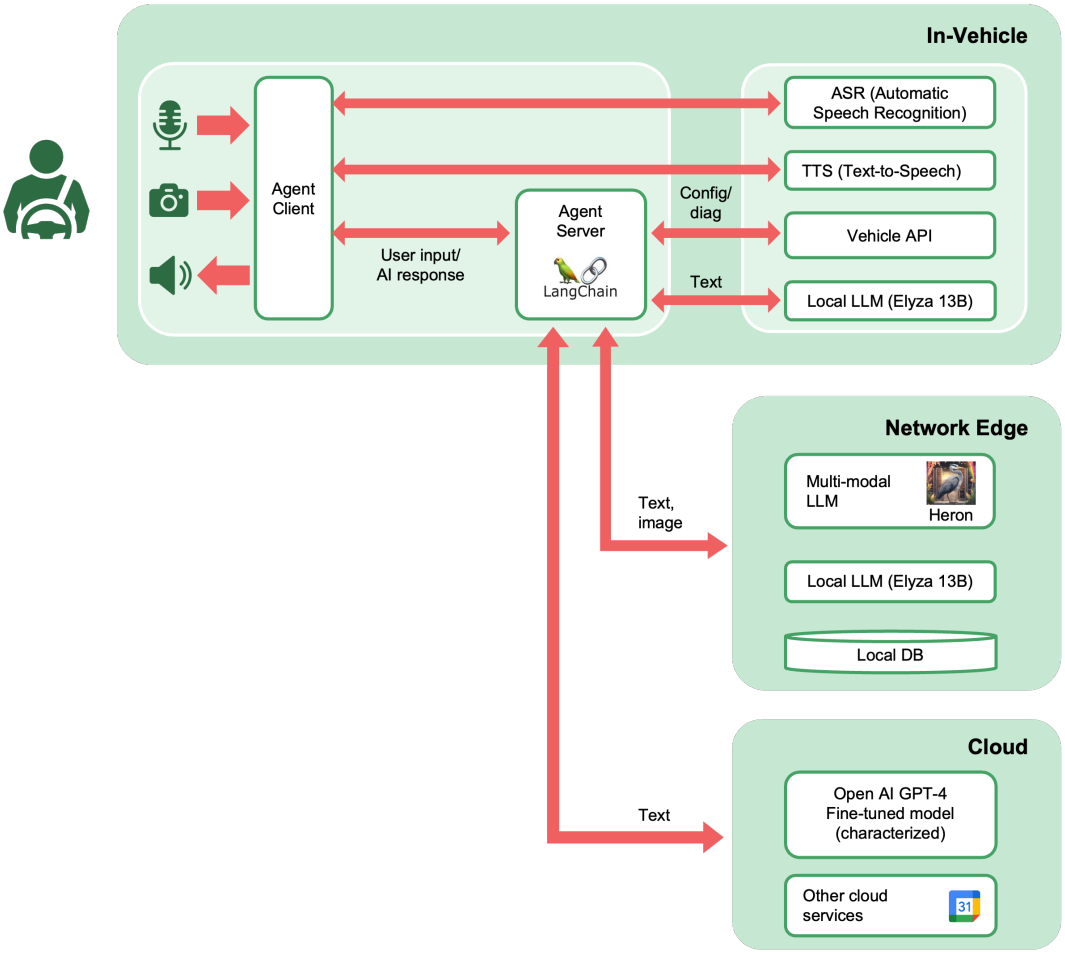


Figure 1 Architecture and implementation overview

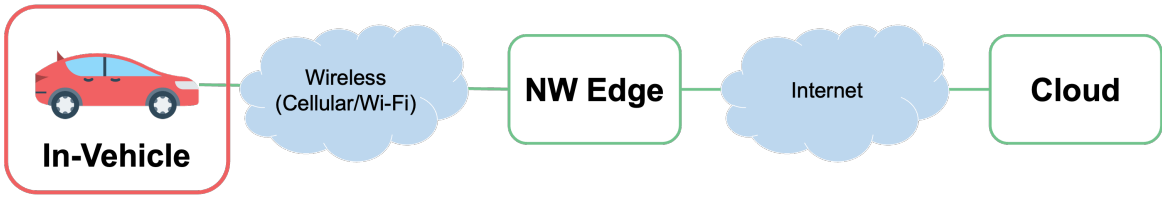
Figure 2 General deployment pattern

One early design challenge for the team was where to place the LLM because each infrastructure location option had both pros and cons. This led to the development of the multi-location solution.

	In-Vehicle	Network Edge	Cloud
Pros	Works anywhere regardless of network connection	More computing resources are available	More computing resources are available (effectively unlimited)
	Zero communication latency	Lower latency	Easier to collaborate with other systems via the Internet
Cons	Available computing resources are limited	Cannot be used without network connection	Cannot be used without network connection
	High communication costs for LLM updates	Slight communication latency	Greater communication latency

The engineering team was very deliberate about their choices in LLM, depending on the location and the nature of the query. Each instance below includes information about the LLM.

Scenario #1: In-vehicle LLM



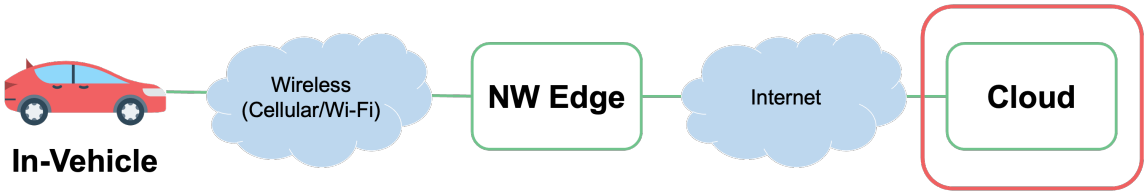
For very simple conversations and interactions about driving or the vehicle, the team chose Elyza, an open Japanese LLM.

The team decided to run it on the in-vehicle system because it can operate smoothly with a responsive speed, especially if the vehicle has a high-performance graphics processing unit (GPU), as many premium-class vehicles are expected to do.

Example interactions included:

1. “I’m sleepy. Let’s chat.”
2. “Set the temperature to 22 degrees Celsius.”
3. “Drive at a speed of 80 kilometers per hour.”

Scenario #2: Cloud LLM

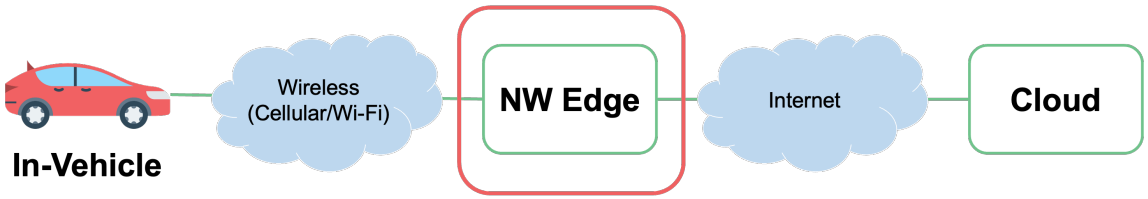


For the cloud scenario, the team chose OpenAI's GPT-4 API, because it was easy to fine-tune the LLM via the API to create characters and integrate the LLM with other cloud systems using retrieval augmented generation (RAG) architecture. To interact with AI characters, you need LLMs that can retain the character's conversation history and audio data. Using cloud LLMs works for this application because there would be no operational impact on the vehicle or driver if the network connection was lost during this entertainment type of interaction. An additional advantage lies in the ability for the LLM to be used in a way that allows switching between AI characters during conversation.

Example interactions included:

1. Interactions with other systems like, "What's my meeting schedule for today?"
2. Conversations with character AIs, for example, "Tell me a funny joke."
3. Advanced conversations like, "Let's discuss the technology behind AI agent systems for vehicles using LLM."

Scenario #3: Network Edge LLM



For use cases involving interactions with in-vehicle and other cameras, the team used a multi-modal LLM with access to the vehicle's onboard camera data, located at the network edge.

The POC team chose to run the system on the network edge for a couple of key reasons. First, while the team may eventually integrate it into vehicles, the current computing power in most cars isn't strong enough to handle the required high-performance multi-modal LLM. Second, sending live camera footage from vehicles to the AI system would use up too much network bandwidth. To avoid overloading the network, we decided to process the data closer to the source, on the network edge, rather than in the cloud.

For standard vehicles without powerful graphics processing, the team showed that the system can be accessed through the network edge. While it could work in the cloud, using the edge is faster, giving quicker responses when interacting with the AI agent.

Example interactions included:

1. "What's that tower over there?"
2. "What's the traffic like based on the cameras you can access?"

The LLM that the team chose for this use case was Heron, an open multimodal LLM developed by Turing. Heron was developed for use by in-vehicle cameras in autonomous driving and driver assistance applications, and it excels at explaining road conditions.

To process related simple queries, the network edge was also equipped with Elyza (explained above).

Proof of Concept Results

In this PoC, the engineering team successfully built a prototype system that integrated multiple LLMs across different settings. The system demonstrated improved conversational abilities, such as flexible command handling, more advanced dialogues, and real-time vehicle diagnostics and adjustments.

The results showed that AECC technologies provided a solid foundation for deploying AI agents efficiently, leading to a better user experience and smoother operations.

Next Steps

The team plans to design a comprehensive large language model operations (LLMOps) platform that will handle everything from LLM training and inference to model distribution.

This platform will leverage both cloud and edge resources to fine-tune the LLMs, maximizing their performance and adaptability. Additionally, the system will incorporate advanced features like retrieval-augmented generation (RAG), with careful consideration of data storage, update methods, and overall architecture. These developments promise to push the boundaries of AI capabilities and deliver even more powerful and efficient solutions.

References

1. https://voicebot.ai/wp-content/uploads/2020/02/in_car_voice_assistant_consumer_adoption_report_2020_voicebot.pdf