мовних моделей у поєднанні з зовнішніми інструментами та RAG-підходом для автоматизації процесу навчання.

Література:

1. OpenAI. (2025). OpenAI API documentation – Building agents. Tools. URL: https://platform.openai.com/docs/guides/agents#tools
2. OpenAI. (2025). OpenAI API documentation – Retrieval-augmented-generation. URL: https://platform.openai.com/docs/guides/optimizing-llm-accuracy/retrieval-augmented-generation-rag.
3. Gradio documentation – Interface. URL: https://www.gradio.app/docs/gradio/interface

UDC 004.42

Yevhen Berladyniuk, Ph.D Assoc. Prof. Mykola Kozlenko
Vasyl Stefanyk Precarpathian National University, Ukraine

PERFORMANCE ANALYSIS OF MICROSERVICE ARCHITECTURE

Microservices embody a paradigmatic shift toward modularity, scalability, and flexibility in software development [1, p. 52]. At the very core of microservice architecture lies the principle of decomposition, breaking down complex applications into a network of smaller, independent services [2].

The main principles of microservice infrastructure are as follows: modularity, service independence, decentralized data management, automation of deployment processes and service lifecycle management, and the distributed nature of microservices [3]. Overall, microservice architecture provides high scalability, flexibility, and reliability; however, it is accompanied by significant management complexity, increased infrastructure costs, and the need for a well-thought-out approach to inter-service communication..

Designing a microservice architecture involves the use of concepts and methodologies that ensure effective distribution of functionality among services and optimal interaction between them [4]. The choice of architectural style is determined by the specifics of business logic, performance requirements, and operational characteristics. This report examines the key principles of microservice architecture, its advantages and disadvantages, as well as approaches to designing microservice-based systems.

The application of microservice and monolitic architectures to energy efficiency monitoring systems [4], [5], robotics [6], and remote learning systems [7] is presented.

Referetces

1. V. Kyselevych, O. Usata, Y. Sikora, D. Verbivskyi, and D. Ivanov, "Microservice architecture: advantages and disadvantages of its practical application," *Information Technology: Computer Science, Software Engineering and Cyber Security*, vol. 2, pp. 50-59, 2024. doi: 10.32782/IT/2024-2-7

2. J. Lewis and M. Fowler, "Microservice trade-offs," *martinfowler.com*, Nov. 2021. [Online]. Available: https://martinfowler.com/articles/microservice-trade-offs.html

3. P. Tupsakhare, "Monolithic vs. microservices: analyzing architectural paradigms in modern software development," *Journal of Scientific and Engineering Research*, vol. 6, no. 2, pp. 299-303, 2019.

4. I. Nuñez *et al.*, "Design of a microservices-based architecture for residential energy efficiency monitoring," *International Journal of Electronics and Telecommunications*, vol. 70, no. 4, pp. 1089-1098, 2024

5. M. Kozlenko and M. Kuz, "Joint capturing of readouts of household power supply meters," *2016 13th International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET)*, Lviv, 2016, pp. 755-757, doi: 10.1109/TCSET.2016.7452172

6. M. Kozlenko, "The interference immunity of the telemetric information data exchange with autonomous mobile robots," *Naukovyi Visnyk Natsionalnoho Hirnychoho Universytetu*, no. 1, pp. 107-113, 2015

7. M. Dutchak, M. Kozlenko, I. Lazarovych, N. Lazarovych, M. Pikuliak, and I. Savka, "Methods and software tools for automated synthesis of adaptive learning trajectory in intelligent online learning management systems," in *Innovations in Smart Cities Applications Volume 4*, in Lecture Notes in Networks and Systems, vol. 183, Karabuk, Turkey, Oct. 07-09, 2020, pp. 206-217. doi: 10.1007/978-3-030-66840-2_16

UDK: 004.896:004.93:629.113:004.774

Yevhen Biezhyn Student, Ph.D.; Assoc. Prof.; Assoc. Prof. Tetiana Sichko
Vasyl’ Stus Donetsk National University, Ukraine

INTEGRATION OF ARTIFICIAL INTELLIGENCE INTO VISION SYSTEMS FOR SELF-DRIVING CARS: CHALLENGES OF DATA TRANSMISSION IN URBAN NETWORKS

The age of self-driven travel is impossible without the integration of artificial intelligence (AI) and vision systems as the "eyes" of autonomous vehicles. Yet, city environments, where the density of obstacles, object movement, and human interaction are highest, place new requirements on technological development. Data transmission in such environments becomes a complex problem that combines technical limitations, cyber risks, and ethical dilemmas.

The role of artificial intelligence in the interpretation of visual information

Modern vision systems for self-driving cars are based on deep neural networks that can analyse images from cameras, lidars, and other sensors in real time. For example, image segmentation algorithms are good at distinguishing between pedestrians, traffic lights, and road signs, while object detection in real time helps maintain safety on the roads. However, the accuracy of such systems is directly related to the data transmission rate between the car, infrastructure, and cloud services. [1]

Theoretical aspects of data transmission challenges

- Latency as a risk factor

In an urban environment where split-second decisions are made, delays in data transmission can be fatal. For example, if a vision system detects a pedestrian, but the information is delayed due to communication interference, the car will not be able to stop in time. Studies show that the maximum permissible latency for self-driving cars should not exceed 10 ms for safety-critical subsystems, which requires the development of technologies such as MEC (Multi-access Edge Computing) to process data directly at the point of receipt. [2]

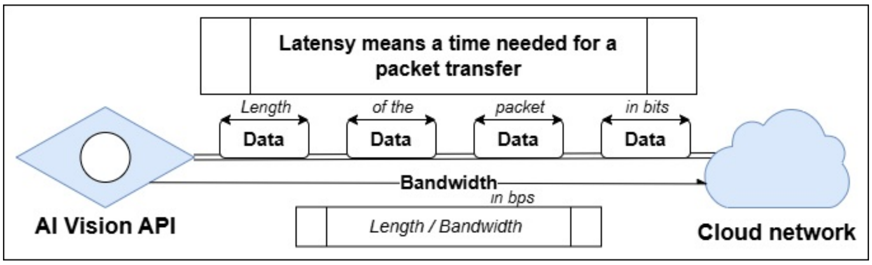

Figure 1: Dependence of data transfer rate on network bandwidth and size of transferred data

- Infrastructural differences

Even in sophisticated cities, the deployment of 5G networks is spotty. Blind spots where autonomous cars lose contact are created by low-coverage zones such as underpasses or high-density