Baština Akademije nauka i umjetnosti Bosne i Hercegovine

SMART MANUFACTURING ON THE HORIZON: A Conceptual Shift from Industry 4.0 to Industry 5.0: Industrija

Karabegović, Isak

2026

Academy of Sciences and Arts of Bosnia and Herzegovina

https://bastina.anubih.ba/handle/123456789/875

Preuzeto s Baštine Akademije nauka i umjetnosti Bosne i Hercegovine

https://bastina.anubih.ba/

DOI: 10.5644/PI2026.225.04

Smart Manufacturing Enabled By LLM AI Agents and Retrieval-Augmented Generation

Mijodrag Milošević*[1]

Abstract: *As smart manufacturing evolves beyond automated execution toward knowledge-based decision support, the effective integration of multidisciplinary knowledge has become a key challenge. Retrieval-Augmented Generation (RAG)-based Large Language Models (LLMs) provide a practical approach to incorporating external knowledge into decision-making processes. They are increasingly being applied in on-site manufacturing environments, where decision-making typically requires frequent human–AI interactions and rapid responses. This paper presents a framework for applying RAG-based LLM AI agents in smart manufacturing environments and discusses their potential to support intelligent, context-aware decision-making.*

Keywords: *Smart Manufacturing, Artificial Intelligence, Large Language Model – LLM, Retrieval-Augmented Generation – RAG, AI Agent*

1. Introduction

A Large Language Model (LLM) is a type of artificial intelligence (AI) trained on a large-scale data set. It is designed to understand and generate human language by learning statistical patterns from textual data. At its core, an LLM is based on deep learning techniques. In general, the larger and more diverse the training data set, the more advanced its Natural Language Processing (NLP) capabilities. NLP is a branch of artificial intelligence that enables computers to understand, interpret, and generate human language [1]. It is a multidisciplinary field that combines computer science, artificial intelligence, and linguistics. The relationship between these disciplines illustrated in Figure 1. By bridging the gap between human communication and machine understanding, NLP enables everyday technologies like virtual assistants, translation tools, and automated chatbots. NLP combines computational linguistics (rule-based language modeling) with machine learning and deep learning. Instead of rigidly programming a computer with every grammar rule, NLP models are trained on big data sets using neural networks to learn statistical patterns and contextual meanings. In smart manufacturing environments, NLP-based AI models can

*[1]University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia
E-mail: mido@uns.ac.rs

automate data collection, extract relevant patterns of system and process behavior, and suggest new strategies and solutions [1].

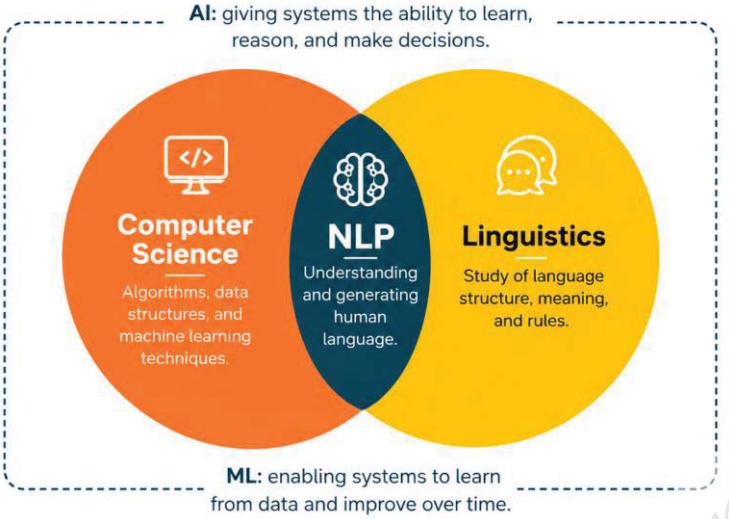


Figure 1. Natural Language Processing (NLP)

In general, AI researchers consider LLMs with billions of parameters to be large language models. Parameters are the internal variables learned during the training process that determine how the model represents and processes information. In general, models with a larger number of parameters exhibit greater representational capacity and are capable of performing more complex language understanding and generation tasks. As LLM architectures continue to evolve, the number of parameters has steadily increased, resulting in significant improvements in model performance and capabilities [2].

2. Context Augmentation Techniques

In today's rapidly evolving AI landscape, organizations are seeking efficient ways to integrate AI solutions that are accurate, secure, and reliable. Although LLMs provide a natural language interface between users and data and are pre-trained on vast amounts of publicly available information, they do not have access to proprietary organizational knowledge [3]. Such knowledge is often organization-specific or domain-specific and is essential for solving specialized tasks. It is typically stored behind APIs, in SQL databases, or within unstructured sources such as PDF documents, technical drawings and tables, as shown in Figure 2.

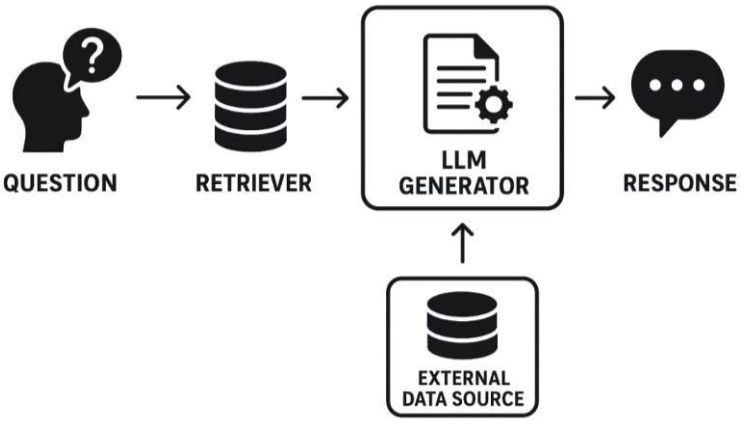


Figure 2. Augmenting the context of an LLM model

Without access to external knowledge, a generative model is limited to producing responses based solely on its parametric knowledge, which is acquired during the training process. Context augmentation enables the integration of external knowledge sources, commonly referred to as non-parametric knowledge, thereby extending the model's effective context and improving the relevance of its responses. Context augmentation is widely used in applications such as:

- Providing more accurate and domain-specific answers to user queries.
- Intelligent chatbots and virtual assistants.
- Document understanding and information extraction.
- Autonomous AI agents capable of reasoning, conducting research, and performing actions.
- Multimodal applications that combine text with images, technical drawings, and other data modalities.

External data sources are typically task-specific and extend beyond the scope of a model's original training data and its parametric knowledge. They are commonly stored in vector databases and may vary considerably in both domain and format. In AI agent systems, this external knowledge base is often regarded as a form of long-term memory that the agent can query whenever additional information is required. Common sources of external knowledge include proprietary organizational databases, legal documents and regulations, scientific literature, technical documentation, and information retrieved from the Web, as shown in Figure 3.



Figure 3. Methods for context augmentation of an LLM model

3. Retrieval-Augmented Generation (RAG)

3.1. RAG Architecture

One of the most important context augmentation methods is Retrieval-Augmented Generation (RAG), which integrates external knowledge with LLMs during the inference phase (Figure 4). This approach enables AI systems to provide up-to-date and accurate information while reducing the risk of hallucinations and incorrect responses. By combining the capabilities of information retrieval with the natural language understanding and generation abilities of LLMs, RAG makes artificial intelligence more reliable, accurate, and practical for real-world applications [4,5]. External data must first be cleaned and transformed into a format that can be processed by the model. Before an LLM can retrieve information from an external knowledge base, the data must be ingested, structured, and indexed. This process is referred to as the indexing (data preparation) phase, which consists of the following steps:

- **Data Loading and Parsing:** Extracting raw data from PDFs, websites, databases, or APIs and converting it into a standardized text format.
- **Chunking:** Dividing large documents into smaller, semantically meaningful segments that fit within the context window of the LLM.
- **Embedding Generation:** Using specialized machine learning models to convert text segments into numerical vector representations (embeddings) that capture their semantic meaning.

- **Storage and Indexing:** Storing these vector representations in a vector database, where they are indexed to enable fast and efficient similarity-based retrieval.

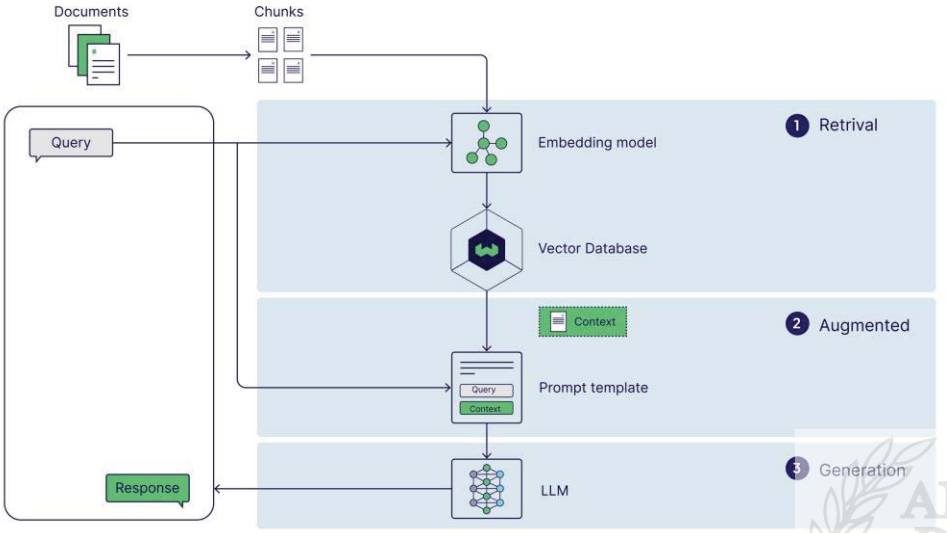


Figure 4. Retrieval-Augmented Generation (RAG) [6]

During the indexing phase, textual or graphical data are transformed from their raw format into vector embeddings through a process known as vectorization. Once the embeddings have been generated, they must be stored in a manner that enables efficient retrieval. In most RAG systems, these embeddings are stored in a vector database, which supports fast and efficient similarity-based retrieval for downstream tasks. This external repository serves as a reusable knowledge base from which relevant information is retrieved during the inference phase (Figure 5).

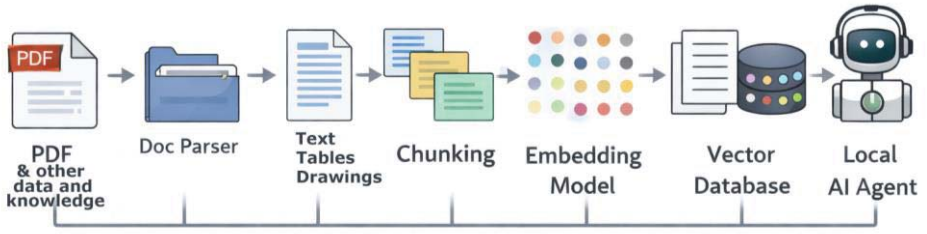


Figure 5. Vectorization of external data and knowledge for context augmentation

After the external data has been encoded and stored, it is ready for retrieval during the inference phase, when the model generates responses to user queries. The inference process consists of three steps: retrieval, augmentation, and generation.

3.2. Retrieval

Based on the user's query, the AI system retrieves relevant information from external knowledge sources, such as documents, databases, or websites. This retrieval process provides the additional context required to generate an accurate response and consists of the following steps:

- **Query Encoding:** The user's query is converted into an embedding within the same vector space as the indexed documents.
- **Vector Search:** The RAG system compares the query embedding with the document embeddings and retrieves the most semantically relevant document chunks.
- **Re-ranking:** An optional specialized model reorders the retrieved chunks according to their semantic relevance, placing the most relevant information at the top of the retrieved context. Although optional, this step is commonly used to improve the overall performance and accuracy of RAG systems.

3.3. Augmentation

The augmentation phase enriches the user's query with relevant information retrieved from external knowledge sources before it is processed by the LLM. During this phase, the retrieved content is selected, filtered, and organized into a structured context that is incorporated into the prompt. This enables the LLM to generate responses based on up-to-date, domain-specific knowledge rather than relying solely on its pre-trained parameters. As a result, the augmentation phase improves response accuracy, reduces hallucinations, and enhances the reliability of the generated output. The original user query is combined with the retrieved context through the following steps:

- **Query Assembly:** The system inserts the retrieved text chunks together with the user's query into a predefined prompt template.
- **Prompt Optimization:** The augmented prompt is refined, compressed if necessary, and formatted to ensure that the LLM effectively interprets and utilizes the provided context when generating the final response.

3.4. Generation

Using the retrieved information, the AI system generates responses that are both accurate and contextually relevant. During the generation phase, the augmented prompt is processed by the LLM, which combines its internal language understanding with the retrieved external knowledge to produce the final response. This enables the model to generate answers that are grounded in up-to-date, domain-specific information while maintaining fluent and coherent natural language generation.
The generation phase focuses on transforming the augmented context into a clear, informative, and well-structured response tailored to the user's query. While the augmentation phase enriches the prompt with relevant external knowledge, the generation phase uses this enriched context to produce the final output. This process typically consists of the following steps:

- **Response Synthesis:** The augmented prompt is submitted to the LLM, which synthesizes the retrieved information with its internal knowledge to generate a coherent and contextually appropriate response.
- **Response Generation:** The LLM produces the final response, leveraging both the retrieved context and its pre-trained knowledge to ensure accuracy, relevance, and fluency.

Figure 6 illustrates an example of an LLM-based AI agent integrated with a RAG architecture for smart manufacturing applications [7].

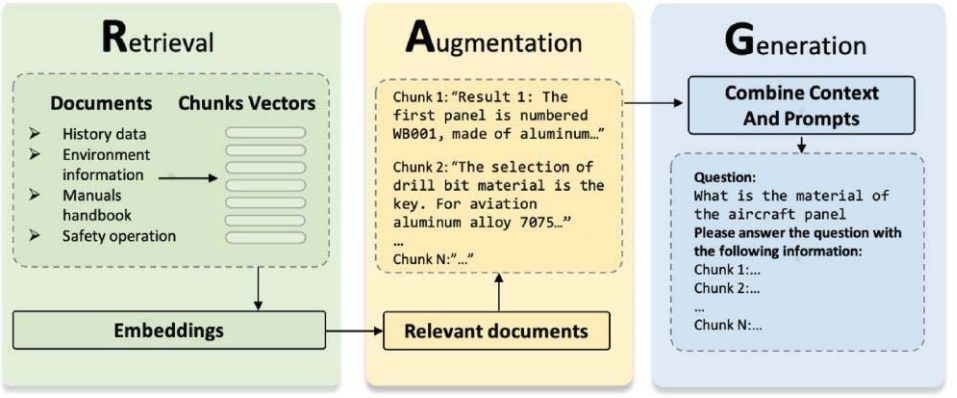


Figure 6. Application of a RAG-based AI agent in Smart Manufacturing framework

4. AI Agentic RAG in Smart Manufacturing

Agentic RAG describes RAG architecture enhanced with AI agents. Unlike conventional RAG systems, AI agents coordinate the retrieval process, manage interactions among system components, and perform additional reasoning and decision-making tasks beyond simple information retrieval and response generation [8]. Rather than relying on a single retrieval step, Agentic RAG employs an iterative workflow in which the agent can retrieve, evaluate, refine, and verify contextual information before generating the final response. This iterative process improves robustness, accuracy, and reliability, making Agentic RAG particularly suitable for complex industrial applications and smart manufacturing environments [9–11].

Figure 7 illustrates the integration of an LLM, a RAG architecture, and an AI agent for the analysis of CAD models and technical documentation. The AI agent orchestrates the entire workflow, beginning with the loading and analysis of the CAD model, followed by the extraction of geometric and technological features [12]. The extracted information is then used to query a vector knowledge base containing embeddings of technical literature, engineering standards, manuals, and previously validated solutions. Finally, the LLM generates a context-aware response that may include recommendations, technical explanations, or references to relevant documentation related to the analyzed CAD model. In this way, the AI agent supports intelligent automation of product analysis and decision-making in a smart manufacturing environment.

As an example of the application of this architecture in smart manufacturing, Figure 8 illustrates a RAG-based anomaly detection system. The system uses RAG to correlate real-time sensor data with historical equipment-specific operational data retrieved from a vector knowledge base. By combining current operating conditions with relevant historical knowledge, the system can identify early signs of anomalies that may not be detected by traditional monitoring methods.

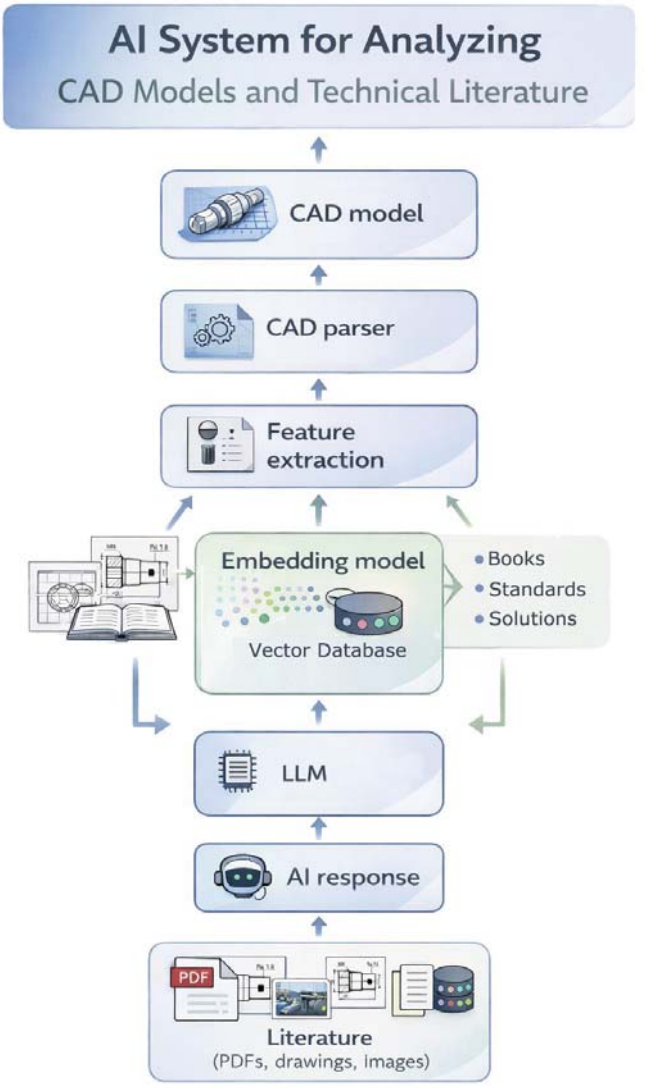


Figure 7. Integration of an LLM–RAG AI agent for CAD model analysis

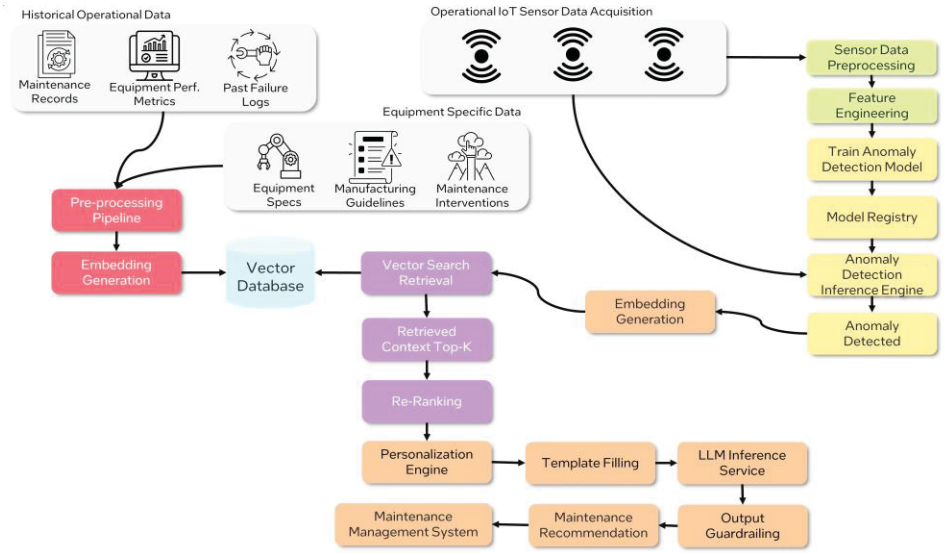


Figure 8. RAG-based Anomaly Detection System [13]

The system continuously monitors equipment data collected from sensors. When an anomaly is detected, it immediately queries the RAG vector knowledge base, which serves as a repository of maintenance records, manufacturer documentation, technical manuals, and historical operational data. This retrieved knowledge is integrated with the LLM to generate context-aware maintenance recommendations and propose an appropriate maintenance strategy.

5. Conclusion

The application of LLMs, enhanced with the RAG context augmentation technique and AI agents, represents a significant step toward the development of intelligent smart manufacturing systems. By integrating advanced natural language understanding, access to relevant domain knowledge, and autonomous decision-making capabilities, these technologies enable more efficient management of manufacturing processes, faster problem solving, improved predictive maintenance, optimized production planning, and enhanced product quality. Although challenges such as reliability, data security, and decision explainability remain, the continuous advancement of LLMs, RAG architectures, and multi-agent systems indicates that these technologies will play a key role in the future of both horizontal and vertical integration in manufacturing, ultimately contributing to the realization of smart manufacturing.

References

[1] Nabavi, S.F., Garmestani, H., Fekri, F. (2025), AI-powered language models for alloy design and laser-based manufacturing: A review of NLP applications in materials science, *Journal of Manufacturing Processes*, Vol. 156 (Part B), pp. 86–120, https://doi.org/10.1016/j.jmapro.2025.11.035

[2] Annepaka, Y., Pakray, P. (2025), Large language models: a survey of their development, capabilities, and applications, *Knowledge and Information Systems*, Vol. 67, pp. 2967–3022, https://doi.org/10.1007/s10115-024-02310-4

[3] Freire S.K.,, Wang, C., Foosherian, M., Wellsandt, S., Ruiz-Arenas, S., Niforatos, E. (2024), Knowledge sharing in manufacturing using LLM-powered tools: user study and model benchmarking, *Frontiers in Artificial Intelligence*, Vol. 7, https://doi.org/10.3389/frai.2024.1293084

[4] Zhao, S., Shao, Y., Huang, Y., Song, J., Wang, Z., Wan, C., Ma, L. (2024), Understanding the Fundamental Design Decisions of Retrieval-Augmented Generation Systems, *ACM Transactions on Software Engineering and Methodology*, https://doi.org/10.1145/3802824

[5] Lee, G., Kim, S., Byun, Y., Lee, J., Choi, H.J.(2026), A Review of Knowledge Base Construction Strategies for LLM-based Intelligent Decision Support System in Material Selection and Process Planning, *International Journal of Precision Engineering and Manufacturing-Green Technology*, Vol. 13, pp. 1393-1430, https://doi.org/10.1007/s40684-026-00890-w

[6] https://weaviate.io/blog/introduction-to-rag, accessed: June 2026.

[7] Dong, W., Li, D., Ji, Yuchen, Chen, H., Liu, S., Mab, Z., Hao, F., Ji, Yuqi, Xing, H., Zheng, P. (2025), Towards a next-generation LLM empowered low-code programming industrial robotic system for human-centric smart manufacturing, *Journal of Manufacturing Systems*, Vol. 83, pp. 675–686, https://doi.org/10.1016/j.jmsy.2025.10.012

[8] https://weaviate.io/blog/what-is-agentic-rag, accessed: June 2026.

[9] Li, M., Zhou, Q., Li, W., Qu, T., Yang, M., Jiang, P. (2025), A4PS: Agentic AI-assisted advanced planning and scheduling with large language models for smart manufacturing, *Journal of Manufacturing Systems*, Vol. 85, pp. 207-226, https://doi.org/10.1016/j.jmsy.2026.01.003

[10] Liu, J., Zhou, X., Yu, S., Duan, X., Wang, J., Wen, H., Zhong, J., Dou, Z. (2026), LLM Agent-driven Automated DFA Assessment with Fine-tuning and AAS-based RAG, *Robotics and Computer-Integrated Manufacturing*, Vol. 102, Art. 103325, https://doi.org/10.1016/j.rcim.2026.103325

[11] Zhou, T, Liu, Y., Kumar, M., Zou, L. (2026), Lightweight query-adaptive RAG framework for knowledge support in smart manufacturing,

Journal of Manufacturing Systems, Vol. 86, pp. 774–789, https://doi.org/10.1016/j.jmsy.2026.04.016

[12] Zhou, J., Camba, J.D. (2025), The status, evolution, and future challenges of multimodal large language models (LLMs) in parametric CAD, *Expert Systems with Applications*, Vol. 282, https://doi.org/10.1016/j.eswa.2025.127520

[13] Alvarez, E., Transforming Manufacturing with RAG: Delivering NextGen Equipment Maintenance, https://eduand-alvarez.medium.com/, accessed: June 2026.

ANU BiH АНУ БиХ