УДК 004.438

V. O. Slobodianiuk
V. V. Chopliak

ARCHITECTURAL PATTERNS FOR INTEGRATING LARGE LANGUAGE MODELS INTO JAVA SPRING BOOT WEB APPLICATIONS

Vinnytsia National Technical University

Анотація

Публікація досліджує ефективні підходи до інтеграції великих мовних моделей (LLM) у веб-додатки, побудовані на базі фреймворку Java Spring Boot. Здійснено аналіз архітектурних патернів для генерації динамічного контенту та текстових пояснень у реальному часі. Встановлено, що використання асинхронної взаємодії через REST API на рівні сервісного шару дозволяє оптимізувати час відгуку системи та уникнути блокування основного потоку. Такий підхід забезпечує високу масштабованість та надійність при розробці сучасних інтелектуальних систем.

Ключові слова: штучний інтелект, великі мовні моделі, генерація контенту, архітектура програмного забезпечення, Java, Spring Boot, REST API.

Abstract

The publication explores effective approaches to integrating Large Language Models (LLMs) into web applications built on the Java Spring Boot framework. An analysis of architectural patterns for generating dynamic content and real-time text explanations is carried out. It is established that using asynchronous communication via REST APIs at the service layer level optimizes the system's response time and prevents main thread blocking. This approach ensures high scalability and reliability when developing modern intelligent systems.

Keywords: artificial intelligence, large language models, content generation, software architecture, Java, Spring Boot, REST API.

Introduction

The rapid advancement of Artificial Intelligence (AI) has fundamentally changed the landscape of modern web development. Integrating Large Language Models (LLMs) into standard web applications is no longer an experimental feature but a core requirement for projects aimed at delivering personalized user experiences, automated predictions, and dynamic contextual explanations [1]. The Java ecosystem, particularly the Spring Boot framework, remains a dominant force in backend development due to its robustness and extensive ecosystem [2]. The primary challenge developers face today is establishing an efficient, non-blocking architectural bridge between the deterministic environment of a Spring Boot backend and the probabilistic, often latency-prone nature of third-party AI APIs.

Research results

Integrating an LLM into a typical Spring Boot application requires a clear separation of concerns to prevent system degradation. Direct, synchronous calls to external AI services within the main HTTP request thread often lead to significant performance bottlenecks, as generating a detailed text explanation via AI can take several seconds [3].

To mitigate this, a multi-layered architectural approach is recommended. First, the application should implement a dedicated "AI Service Layer" responsible solely for constructing prompts and communicating with the LLM API (e.g., via Spring WebClient). This layer must isolate the external dependency, ensuring that if the AI service experiences downtime, the core application functionality remains accessible.

Secondly, for generating complex predictions or lengthy explanations, an asynchronous pattern using @Async in Spring Boot [4], coupled with message queues (like RabbitMQ) or WebSockets, proves to be the most efficient. When a user requests an AI-generated explanation, the backend immediately returns a confirmation response and processes the AI request in the background. Once the external LLM returns the generated text, the application updates the database and notifies the client via WebSocket or Server-Sent Events (SSE). This pattern not only improves the perceived performance of the system but also allows for better error handling and retry mechanisms without frustrating the end user [3].

Conclusion

The conducted analysis demonstrates that directly coupling AI integration logic with standard synchronous request-response cycles in Java Spring Boot applications is suboptimal. Implementing a dedicated, asynchronous service layer for LLM interactions ensures system resilience, optimizes resource utilization, and provides a seamless user experience when generating dynamic content. As AI components become standard in web development, adopting these non-blocking architectural patterns will be crucial for maintaining high-performance backend systems.

REFERENCES

1. Tunio M. Z. et al. Integrating Large Language Models into Software Development: A Systematic Review. *IEEE Access*, 2023.
2. Walls C. Spring in Action. 6th ed. Shelter Island: Manning Publications, 2022. 536 p.
3. Richardson C. Microservices Patterns: With examples in Java. 1st ed. Shelter Island: Manning Publications, 2018. 520 p.
4. Spring Framework Documentation: Asynchronous Execution. URL: https://docs.spring.io/spring-framework/reference/integration/scheduling.html.
5. Kravchenko K., Ketsyk-Zinchenko U., Suduk I., Nykyporets S., Cherednychenko V. Effectiveness of online platforms in developing language skills of higher education students. *Revista Eduweb*. 2025. 19(3). P. 303-314. DOI: https://doi.org/10.46502/issn.1856-7576/2025.19.03.19.
6. Nykyporets S. S., Herasymenko N. V., Chopliak V. V. Developing digital language competence as a factor of competitiveness of future master's degree holders in power engineering in the digital economy. *Artificial intelligence as a tool to protect the economy disinformation: innovative solutions and international practices: collective monograph* / edited by O. H. Cherep. Riga, Latvia : Baltija Publishing, 2025. 258 p. P. 140-193. DOI: https://doi.org/10.30525/978-9934-26-586-0.
7. Ibrahimova L. V., Nykyporets, S. S. Information security in the global context: linguistic perspectives and the role of English. *International security studios: managerial, technical, legal, environmental, informative and psychological aspects.* International collective monograph. Volume II. ISAP, Research and Education. 2025. 436 p., P. 321-345. DOI: https://doi.org/10.5281/zenodo.15356365.
8. Kot S. O., Nykyporets S. S. Utilization of artificial intelligence in enhancing English language proficiency in tertiary education. *Science and Education in the Third Millennium: Information Technology, Education, Law, Psychology, Social Sphere, Management: International collective monograph*. Institute of Public Administration Affairs, Lublin, Polska, 2024. Chap. 10. P. 250-274. DOI: https://doi.org/10.5281/zenodo.11279390.
9. Nykyporets S. S., Kot S. O., Boiko Yu. V., Melnyk M. B., Chopliak V. V. Advanced integration of virtual information environments (VIEs) in contemporary educational methodologies. *Society and national interests. Series «Education/Pedagogy»*. 2024. No. 4(4). Pp. 139-154. https://doi.org/10.52058/3041-1572-2024-4(4)-139-154.

Слободянюк Вадим Олександрович – студент групи 4ПІ-24б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: vadymsl2007@gmail.com.

Науковий керівник: ***Чопляк Вікторія Володимирівна*** – викладач англійської мови, кафедра іноземних мов, Вінницький національний технічний університет, м. Вінниця, e-mail: nikavnuchkova@gmail.com.

Vadym O. Slobodianiuk – Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: vadymsl2007@gmail.com.

Scientific supervisor: ***Victoria V. Chopliak*** – teacher of English, Foreign Languages Department, Vinnytsia National Technical University, Vinnytsia, e-mail: nikavnuchkova@gmail.com.