Industrial Transport of Kazakhstan
ISSN 1814-5787 (print)
ISSN 3006-0273 (online)
Vol. 23. Is. 2. Number 90 (2026). Pp. 83–95
Journal homepage: https://prom.mtgu.edu.kz
https://doi.org/10.58420/ptk/2026.90.02.007

LEARNLY PLATFORM BASED ON MACHINE LEARNING FOR ADAPTIVE AND PERSONALIZED LEARNING

K.B. Baltabayev[1], K.B. Tussupova[1]*, Igor Fernández Plazaola[2]
[1]al-Farabi Kazakh National University, Almaty, Kazakhstan;
[2]Universitat Politècnica de València, Valencia, Spain.
E-mail: kamshat.tusupova@kaznu.edu.kz

K.B. Baltabayev — student, al-Farabi Kazakh National University, Almaty, Kazakhstan
E-mail: q.baltabayev@icloud.com, https://orcid.org/0009-0004-1342-4379;
K.B. Tussupova — PhD, Associate Professor, al-Farabi Kazakh National University, Almaty, Kazakhstan
E-mail: kamshat.tusupova@kaznu.edu.kz, https://orcid.org/0000-0002-5254-3432;
Igor Fernández Plazaola — PhD, Director Master in Building Engineering, Building Egineering Faculty, Universitat Politècnica de València, Valencia, Spain
E-mail: iplazaola@doe.upv.es, 0000-0003-2382-5075.

Abstract. This study examines the application of machine learning to personalized learning in interactive platforms through the example of Learnly. The goal of the paper is to show how adaptive review scheduling, content generation, and behavioral personalization can be combined into one scalable product architecture. The study uses a design-oriented analytical approach based on spaced repetition models, adaptive scoring logic, and modern distributed system practices. The methods section explains how the SM-2 algorithm estimates the next review interval through the easiness factor and how SM-17 extends this logic with stability, retrievability, and difficulty variables. The paper also introduces the concept of the Learnly Intelligence Layer as a proprietary adaptive model that combines memory dynamics, engagement signals, and AI-assisted content generation. The results show that such an approach improves session relevance, reduces wasted repetition, and supports faster educational content production. Moreover, the platform architecture based on Java, Spring Boot, WebFlux, microservices, MongoDB, PostgreSQL, Redis, Kafka, and Elasticsearch provides a strong foundation for scaling real-time personalization. The study concludes that product value emerges not from isolated AI features but from the coordinated interaction of mathematical scheduling, learning analytics, generative AI, and resilient system design.

Keywords: machine learning, spaced repetition, Apache Kafka, adaptive platforms, educational AI, personalized learning

For citation: Baltabayev K.B., Tussupova K.B., Igor Fernández Plazaola (2026). Learnly platform based on machine learning for adaptive and personalized learning // Industrial Transport of Kazakhstan. Vol. 23. No. 90. Pp. 83–95. https://doi.org/10.58420/ptk/2026.90.02.007 (In Eng.).

Conflict of interest: The authors declare that there is no conflict of interest.

МАШИНАЛЫҚ ОҚЫТУ НЕГІЗІНДЕГІ АДАПТИВТІ ЖӘНЕ ЖЕКЕЛЕНДІРІЛГЕН ОҚЫТУҒА АРНАЛҒАН LEARNLY ПЛАТФОРМАСЫ

Қ.Б. Балтабаев[1], Қ.Б. Тусупова[1]*, Игорь Фернандес Плазаола[2]
[1]Әл-Фараби атындағы Қазақ ұлттық университеті, Алматы, Қазақстан;
[2]Политехнический университет Валенсии, Валенсия, Испания.
E-mail: kamshat.tusupova@kaznu.edu.kz

Балтабаев Қ.Б. — студент, Әл-Фараби атындағы Қазақ ұлттық университеті, Алматы, Қазақстан
E-mail: q.baltabayev@icloud.com, https://orcid.org/0009-0004-1342-4379;
Тусупова Қ.Б. — PhD, қауымдастырылған профессор, Әл-Фараби атындағы Қазақ ұлттық университеті, Алматы, Қазақстан.
E-mail: kamshat.tusupova@kaznu.edu.kz, https://orcid.org/0000-0002-5254-3432;
Игорь Фернандес Плазаола — PhD, директор магистерской программы по инженерии зданий, факультет строительной инженерии, Политехнический университет Валенсии, Валенсия, Испания.
E-mail: iplazaola@doe.upv.es, https://orcid.org/0000-0003-2382-5075.

Аннотация. Бұл зерттеу Learnly платформасы мысалында интерактивті платформалардағы жекелендірілген оқытуда машиналық оқытуды қолдануды қарастырады. Жұмыстың мақсаты — адаптивті қайталау жоспарлауын, контент генерациясын және мінез-құлыққа негізделген жекелендіруді бір масштабталатын өнім архитектурасына біріктіру мүмкіндігін көрсету. Зерттеу spaced repetition модельдеріне, адаптивті бағалау логикасына және заманауи үлестірілген жүйелер тәжірибесіне негізделген жобалық-аналитикалық тәсілді қолданады. Әдістер бөлімінде SM-2 алгоритмінің easiness factor арқылы келесі қайталау интервалын қалай бағалайтыны және SM-17 алгоритмінің бұл логиканы тұрақтылық (stability), еске түсіру ықтималдығы (retrievability) және қиындық (difficulty) айнымалылары арқылы қалай кеңейтетіні түсіндіріледі. Сонымен қатар, мақалада Learnly Intelligence Layer тұжырымдамасы енгізіледі — бұл жады динамикасын, пайдаланушы белсенділігі сигналдарын және жасанды интеллект көмегімен контент генерациясын біріктіретін меншікті адаптивті модель. Нәтижелер мұндай тәсілдің оқу сессияларының өзектілігін арттыратынын, артық қайталауларды азайтатынын және білім беру контентін жылдам жасауға мүмкіндік беретінін көрсетеді. Сонымен қатар, Java, Spring Boot, WebFlux, микросервистер, MongoDB, PostgreSQL, Redis, Kafka және Elasticsearch негізіндегі платформа архитектурасы нақты уақыттағы жекелендіруді масштабтауға берік негіз қалайды. Зерттеу қорытындысы бойынша, өнімнің құндылығы жекелеген AI функцияларынан емес, математикалық жоспарлау, оқу аналитикасы, генеративті жасанды интеллект және сенімді жүйелік архитектураның үйлесімді өзара әрекеттесуінен қалыптасады.

Түйін сөздер: машиналық оқыту, spaced repetition, Apache Kafka, адаптивті платформалар, білім беру AI, жекелендірілген оқыту

Дәйексөздер үшін: Балтабаев Қ.Б., Тусупова Қ.Б., Игорь Фернандес Плазаола (2026). Машиналық оқыту негізіндегі адаптивті және жекелендірілген оқытуға арналған Learnly платформасы // Қазақстан өндіріс көлігі. Том. 23. № 90. 83–95 бет. https://doi.org/10.58420/ptk/2026.90.02.007 (Ағыл. тіл.).

Мүдделер қақтығысы: Авторлар осы мақалада мүдделер қақтығысы жоқ деп мәлімдейді.

ПЛАТФОРМА LEARNLY НА ОСНОВЕ МАШИННОГО ОБУЧЕНИЯ ДЛЯ АДАПТИВНОГО И ПЕРСОНАЛИЗИРОВАННОГО ОБУЧЕНИЯ

К.Б. Балтабаев[1], К.Б. Тусупова[1]*, Игорь Фернандес Плазаола[2]
[1]Казахский национальный университет им. аль-Фараби, Алматы, Казахстан;
[2]Валенсия политехникалық университеті, Валенсия, Испания.
E-mail: kamshat.tusupova@kaznu.edu.kz

Балтабаев К.Б. — студент, Казахский национальный университет им. аль-Фараби, Алматы, Казахстан
E-mail: q.baltabayev@icloud.com, https://orcid.org/0009-0004-1342-4379;
Тусупова К.Б. — PhD, доцент, Казахский национальный университет им. аль-Фараби, Алматы, Казахстан
E-mail: kamshat.tusupova@kaznu.edu.kz, https://orcid.org/0000-0002-5254-3432
Игорь Фернандес Плазаола — PhD, Ғимараттар инженериясы бойынша магистрлік бағдарламаның директоры, Құрылыс инженериясы факультеті, Валенсия политехникалық университеті, Валенсия, Испания
E-mail: iplazaola@doe.upv.es, https://orcid.org/0000-0003-2382-5075.

Аннотация. Данное исследование рассматривает применение машинного обучения в персонализированном обучении на интерактивных платформах на примере Learnly. Цель работы — показать, как адаптивное планирование повторений, генерация контента и поведенческая персонализация могут быть объединены в единую масштабируемую архитектуру продукта. Исследование использует проектно-аналитический подход, основанный на моделях интервального повторения (spaced repetition), адаптивной логике оценки и современных практиках распределённых систем. В разделе методов объясняется, как алгоритм SM-2 оценивает следующий интервал повторения через коэффициент лёгкости (easiness factor), а также как SM-17 расширяет эту логику с использованием переменных стабильности (stability), извлекаемости (retrievability) и сложности (difficulty). В статье также вводится концепция Learnly Intelligence Layer как проприетарной адаптивной модели, объединяющей динамику памяти, сигналы вовлечённости пользователя и генерацию контента с помощью искусственного интеллекта. Результаты показывают, что такой подход повышает релевантность учебных сессий, снижает избыточные повторения и способствует ускоренному созданию образовательного контента. Кроме того, архитектура платформы, основанная на Java, Spring Boot, WebFlux, микросервисах, MongoDB, PostgreSQL, Redis, Kafka и Elasticsearch, обеспечивает надёжную основу для масштабирования персонализации в реальном времени. В заключении отмечается, что ценность продукта формируется не за счёт отдельных AI-функций, а благодаря согласованному взаимодействию математического планирования, аналитики обучения, генеративного искусственного интеллекта и устойчивой архитектуры системы.

Ключевые слова: машинное обучение, spaced repetition, Apache Kafka, адаптивные платформы, образовательный AI, персонализированное обучение

Для цитирования: Балтабаев К.Б., Тусупова К.Б., Игорь Фернандес Плазаола (2026). Платформа Learnly на основе машинного обучения для адаптивного и персонализированного обучения // Помышленный транспорт Казахстана. Т. 23. No. 90. Стр. 83–95. https://doi.org/10.58420/ptk/2026.90.02.007 (На англ.).

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов.

Introduction.

Personalized learning has become one of the most important directions in modern digital education because users no longer expect platforms to behave like passive repositories of lessons or flashcards. They expect interactive systems to estimate what they know, identify what they are likely to forget, and deliver the next action at the right moment (Wozniak, 1990: 1; Wozniak, 2026: 1; Piech et al., 2015: 505–513). In product terms, this means that the value of the platform depends not only on the amount of available content, but also on the quality of the decision logic that organizes the learner's experience (Pandey, Karypis, 2019: 384–389).

Learnly can be described as an interactive educational platform in which adaptive review scheduling, AI-assisted content generation, and engagement management are integrated into one product flow (Kochmar et al., 2022: 323–349; Waladi, Khaldi, Lamarti Sefian, 2023: 4–15; Ismail et al., 2023: 741). This paper considers Learnly not simply as a software application, but as a representative case of how machine learning techniques and memory-based scheduling models can be used to construct a learning environment that is both user-centered and scalable (Madani et al., 2023: 1216). The article focuses on the product logic and the intelligence layer rather than on low-level implementation details.

The purpose of the study is to explain how a personalized learning platform can combine classical spaced repetition, modern adaptive modeling, and scalable system architecture. The objectives are the following: first, to present the mathematical basis of SM-2 and SM-17; second, to show how these models can be interpreted inside an adaptive product such as Learnly; third, to explain what features can participate in a proprietary AI model for prioritizing study content; and fourth, to demonstrate why the selected technology stack supports such an approach at scale.

Materials and methods.

The methodological basis of the paper combines algorithmic analysis and system design analysis. On the algorithmic side, the work relies on spaced-repetition models that formalize how learning intervals should change after each review. On the architectural side, the study examines how these decision rules can be embedded into a distributed learning platform that also includes content generation, analytics, reminders, and search.

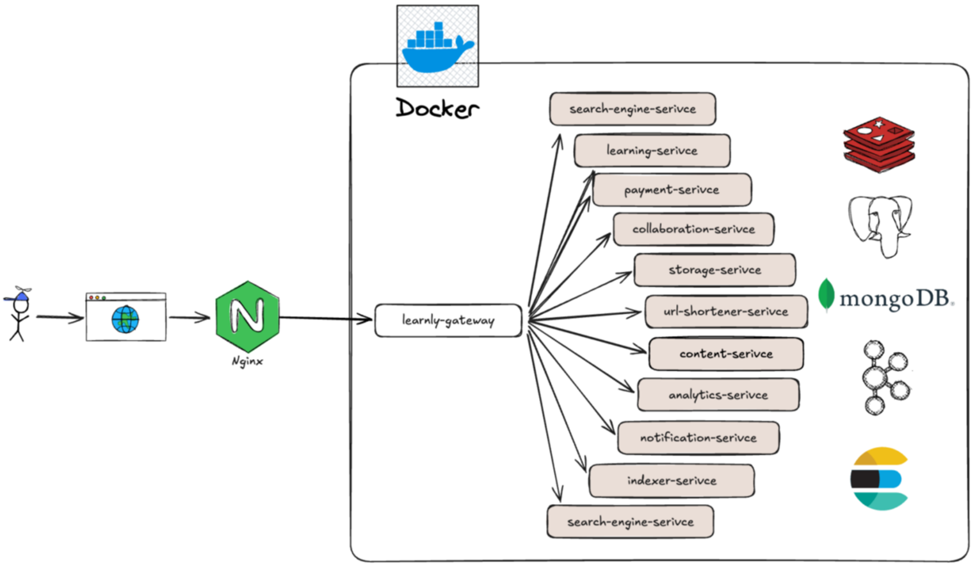

Fig. 1. Architecture of the Learnly distributed learning platform based on microservices and event-driven interaction

Figure 1 illustrates the architecture of the Learnly platform, which is built using a microservices approach. The system is deployed in a containerized environment using Docker, ensuring scalability and isolation of services. Client requests are first processed by Nginx and then routed through the API Gateway, which serves as a single entry point to the system. The gateway distributes requests to the corresponding microservices, including content-service, learning-service, analytics-service, notification-service, and others.

Each service is responsible for a specific domain, following the principles of separation of concerns and independent scalability. The platform integrates multiple storage solutions: PostgreSQL for relational data, MongoDB for flexible document storage, Redis for caching, and Elasticsearch for search and indexing.

Inter-service communication is performed both synchronously (via REST APIs) and asynchronously (via message broker Kafka), enabling event-driven interaction and improving system resilience.

Special attention is given to the content-service, which integrates AI-based generation modules, enabling automatic creation of learning materials based on user input. This architecture allows the platform to support key features such as content generation, spaced repetition learning, analytics, notifications, and search in a scalable and maintainable manner.

Таблица 1 – Technology stack of the Learnly platform

Layer	Technology
Backend	Java 17, Spring Boot
Reactive layer	Spring WebFlux
Messaging	Apache Kafka
Databases	PostgreSQL, MongoDB
Cache	Redis
Search	Elasticsearch
Deployment	Docker

Furthermore, the architecture is designed with production-grade reliability and scalability in mind. Horizontal scaling is achieved through stateless service design and container orchestration, allowing individual services to scale independently under varying load conditions. The use of asynchronous messaging via Kafka decouples services and prevents cascading failures, ensuring fault tolerance and system resilience. Additionally, caching layers (Redis) and search indexing (Elasticsearch) significantly reduce latency and improve response times for high-frequency operations. This combination of design decisions enables the platform to handle large-scale user activity while maintaining high availability, performance, and extensibility for future feature integration.

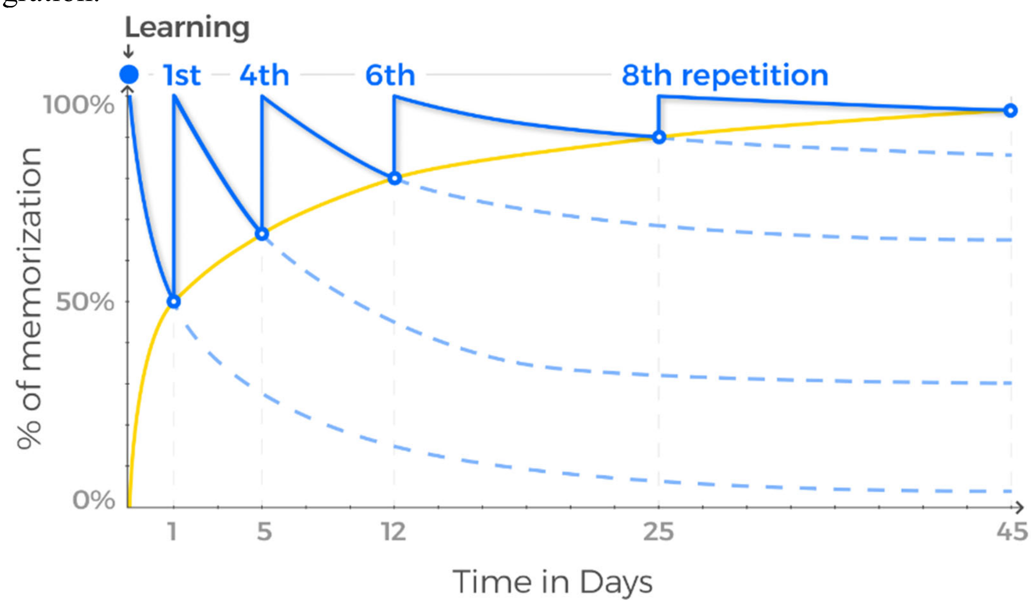

Fig. 2. Ebbinghaus forgetting curve and its application in spaced repetition algorithms

As shown in Figure 2, memory retention decreases exponentially over time without reinforcement. This phenomenon highlights the necessity of scheduling reviews at optimal intervals to prevent information loss. To address this challenge, spaced repetition algorithms have been developed. One of the most widely used approaches is the SM-2 algorithm, which adapts review intervals based on the quality of user responses. While SM-2 provides a simple and efficient mechanism for scheduling repetitions, it does not explicitly model the underlying memory processes. To overcome these limitations, more advanced models such as SM-17 have been introduced. These models introduce a more detailed representation of memory dynamics by separating stability, retrievability, and difficulty, enabling more accurate prediction of forgetting behavior. In the Learnly platform, these theoretical models are extended with a practical prioritization mechanism that considers user behavior, cognitive load, and learning context to optimize both review scheduling and content delivery.

In practice, the SM-2 algorithm calculates the next review interval based on the easiness factor and the quality of the user's response, allowing the system to dynamically adjust repetition frequency. In contrast, SM-17 introduces a more sophisticated memory model, where stability reflects how long information is retained, retrievability represents the probability of recall at a given moment, and difficulty characterizes the inherent complexity of the learning item. This allows for a more adaptive and personalized learning experience.

To formalize the described memory dynamics and enable quantitative modeling of the learning process, it is necessary to introduce algorithmic approaches that can translate theoretical concepts into practical scheduling strategies. In this context, spaced repetition algorithms provide a computational framework for estimating optimal review intervals based on user interaction and memory state.

The first baseline is the SM-2 algorithm. In its classic form, the review intervals are calculated according to formulas (1) and (2). SM-2 uses the easiness factor to increase or slow the growth of the next interval according to recall quality. This approach remains valuable because it is interpretable, computationally lightweight, and effective in many real learning scenarios.

The second methodological layer uses the logic of SM-17. In contrast to SM-2, this model introduces stability, retrievability, and difficulty as separate dimensions of memory dynamics. Retrievability can be estimated with an exponential decay function, while the next interval is linked to the stability increase function. These relationships are shown in formulas (3) and (4).

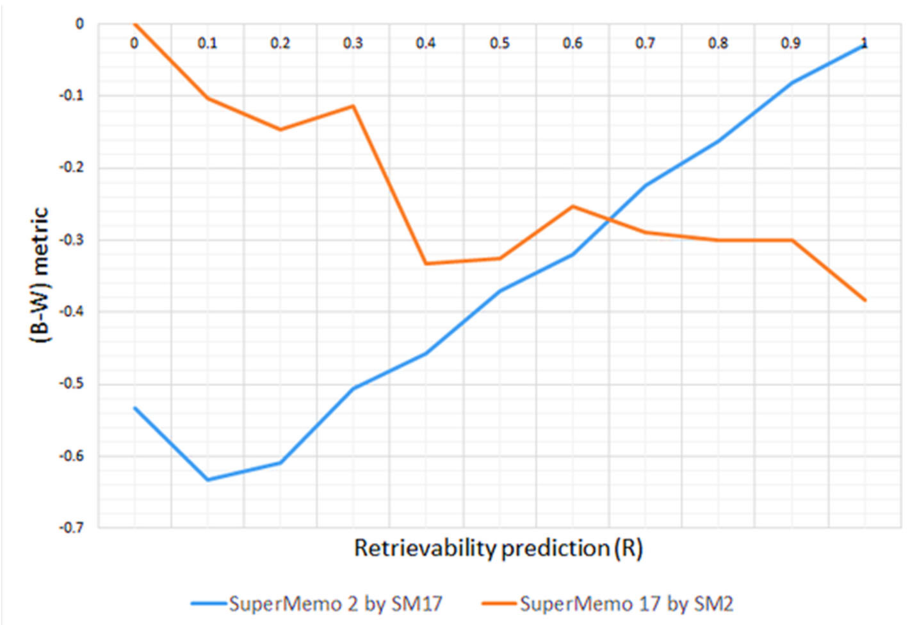

Fig. 3. Comparison of SM-2 and SM-17 memory models

Figure 3 summarizes the key differences between the SM-2 and SM-17 algorithms. While SM-2 relies on a simplified heuristic based on the easiness factor, SM-17 introduces a more structured representation of memory through stability, retrievability, and difficulty.

This comparison highlights the trade-off between computational simplicity and modeling accuracy. SM-2 remains efficient and interpretable, making it suitable as a baseline, whereas SM-17 enables a more precise estimation of memory dynamics, which is critical for adaptive learning systems.

Таблица 2 – Comparison of SM-2 and SM-17 algorithms

Feature	SM-2	SM-17
Model type	Heuristic-based	Memory-based model
Key parameters	Easiness Factor (EF)	Stability, Retrievability, Difficulty
Complexity	Low	High
Adaptivity	Limited	Advanced
Memory modeling	Implicit	Explicit
Accuracy	Moderate	High

To explain how a proprietary adaptive model can operate inside Learnly, the paper also uses a conceptual priority scoring formulation that aggregates forgetting risk, concept importance, lapse probability, session fit, and cognitive load. This scoring view does not replace the memory equations; it extends them into a product-level recommendation mechanism for next-card selection and reminder timing.

Таблица 3 – Formulas of the SM-2 and SM-17 algorithms and the Learnly Intelligence Layer scoring model

Formula	Number
$I(1) = 1;\quad I(2) = 6;\quad \text{for } n > 2:\ I(n) = I(n-1) \times EF$	(1)
$EF' = EF + (0.1 - (5 - q) \times (0.08 + (5 - q) \times 0.02))$	(2)
$R = \exp(-k \times t / S)$	(3)
$Int[n] = S[n-1] \times SInc[D, S, R] \times \ln(1 - rFI/100) / \ln(0.9)$	(4)
$\text{Priority Score} = w_1 \times \text{ForgettingRisk} + w_2 \times \text{ConceptImportance} + w_3 \times \text{LapseProbability} + w_4 \times \text{SessionFit} - w_5 \times \text{CognitiveOverload}$	(5)

Despite their effectiveness, both SM-2 and SM-17 have inherent limitations when applied in real-world product environments. SM-2, while computationally efficient, lacks an explicit representation of memory state and cannot account for contextual learning factors such as cognitive load or user fatigue. SM-17 introduces a richer memory model, but its increased complexity makes real-time computation and large-scale deployment more challenging.

These limitations motivate the need for an additional abstraction layer that can bridge theoretical memory modeling and practical product requirements. In Learnly, this role is fulfilled by the Intelligence Layer, which augments classical algorithms with behavioral signals and system-level constraints.

In formula (1), I(n) denotes the interval after the n-th repetition, and EF denotes the easiness factor. In formula (2), q is the quality of response on a scale from 0 to 5. In formula (3), R describes retrievability, t denotes elapsed time, S denotes memory stability, and k is a decay constant. In formula (4), D is item difficulty, SInc[D,S,R] is the stability increase function, and rFI is the requested forgetting index. Formula (5) summarizes the conceptual scoring logic of the Learnly Intelligence Layer.

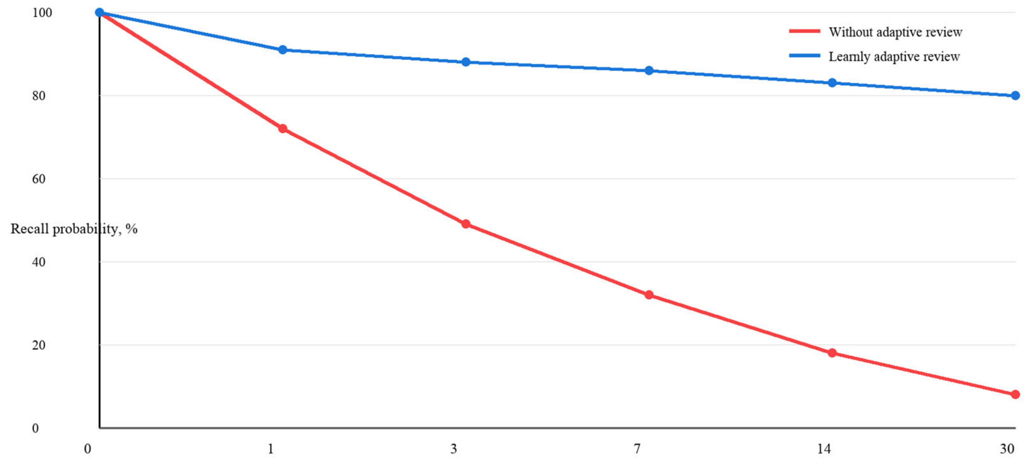

Fig. 4. Retention without adaptive review and with Learnly adaptive review

Figure 4 shows the conceptual difference between passive content exposure and adaptive review strategies. In the non-adaptive scenario, recall probability declines rapidly due to the absence of timely reinforcement, reflecting the classical forgetting curve.

In contrast, the Learnly adaptive review strategy maintains a significantly higher recall level over time by introducing reviews at moments when memory is most susceptible to decay. This results in a slower decline of retention and a more stable long-term memory state.

The key implication of this result is that adaptive intervention does not simply increase repetition frequency, but optimizes the timing of reviews. This reduces unnecessary repetitions while maximizing memory consolidation efficiency.

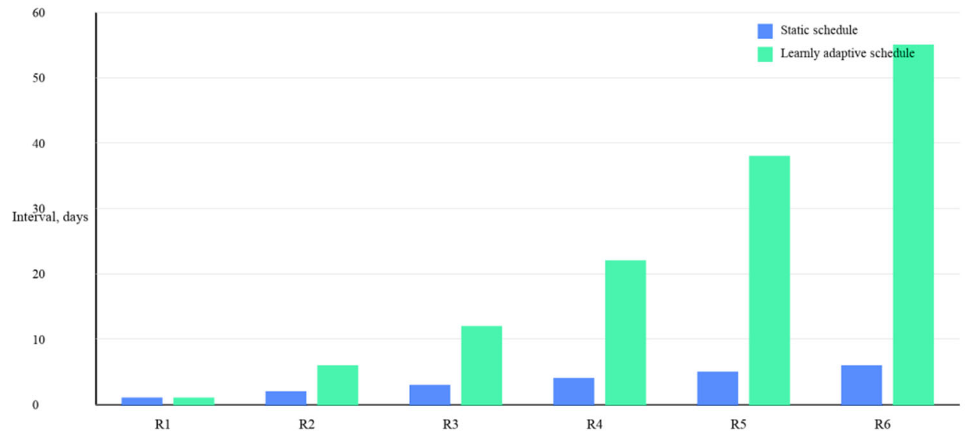

Fig. 5. Growth of review intervals under static and adaptive scheduling

Figure 5 illustrates the difference in interval growth between static scheduling and adaptive spaced repetition. In a static approach, interval progression remains relatively linear and does not account for the learner's actual memory state.

In contrast, the Learnly adaptive schedule demonstrates accelerated interval growth as memory stabilizes. This reflects the system's ability to recognize well-learned material and reduce review frequency accordingly.

At the same time, difficult or unstable items are revisited more frequently, creating a dynamic balance between reinforcement and efficiency. This behavior enables the system to minimize cognitive overload while maintaining high retention levels.

While Figures 4 and 5 focus on the optimization of review scheduling through memory-based models, adaptive learning in Learnly is not limited to deciding when to present content. An equally important dimension is what content should be presented.

In real-world learning environments, scheduling alone is insufficient if the available content does not match the learner's current knowledge state. Therefore, Learnly extends adaptive decision-making beyond interval optimization by incorporating AI-based content generation. This allows the system to dynamically produce learning materials that align with the learner's context, difficulty level, and learning objectives.

The following section describes how this capability is implemented through a machine learning pipeline that transforms user input into structured educational content.

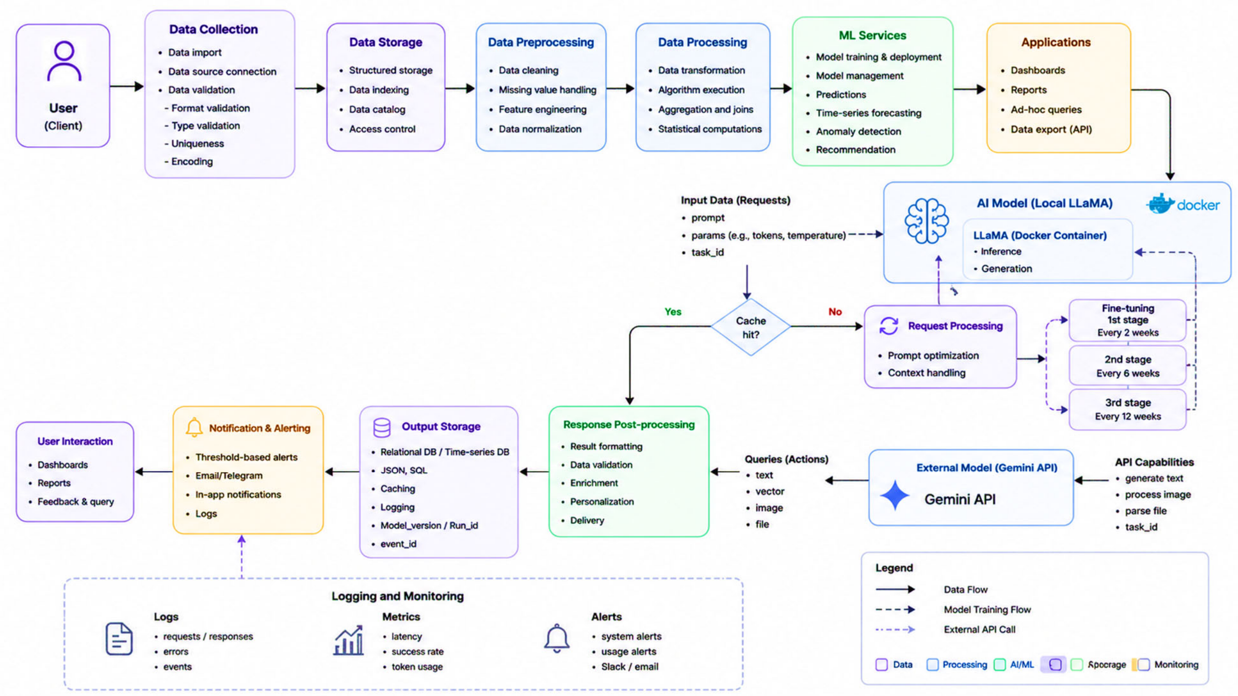

Fig. 6. AI-based content generation pipeline in the Learnly platform

Figure 6 illustrates the end-to-end machine learning pipeline that powers adaptive content generation in the Learnly platform. Unlike traditional systems that rely on static content, Learnly treats generation as a dynamic, model-driven process where user input is continuously transformed into semantically structured learning material.

The pipeline begins with multi-dimensional user input, including topic, difficulty level, language, and contextual constraints. This data is processed through a dedicated prompt engineering layer, which performs context aggregation, semantic enrichment, and instruction shaping. As a result, the system constructs high-quality, task-specific prompts that guide the model toward structured educational output rather than generic text generation.

At the core of the system is a locally deployed transformer-based large language model (LLaMA), running in a containerized environment. The model operates in an inference-optimized setup, where generation is controlled through dynamic parameter tuning (temperature, top-k, top-p, max tokens), enabling fine-grained control over creativity, determinism, and response consistency.

A key aspect of the architecture is its resilience and adaptability. The generation process is wrapped in a fault-tolerant execution layer that includes retry strategies with exponential backoff and quality-aware fallback routing. When the primary model fails to meet quality thresholds or latency constraints, the system automatically switches to an external model (e.g., Gemini API), ensuring uninterrupted service and consistent output quality.

Beyond generation, Learnly applies a structured post-processing pipeline that transforms raw model output into normalized, production-ready learning artifacts. This includes semantic validation, deduplication, formatting, and transformation into question–answer flashcards aligned with the platform's learning model.

Crucially, machine learning in Learnly is not limited to text generation. The generated content is tightly integrated with adaptive scheduling algorithms (SM-2 and SM-17) and a proprietary prioritization layer, which uses behavioral signals such as forgetting risk, concept importance, and cognitive load. This enables the system to not only generate content but also decide when and how it should be presented to the user.

The platform continuously collects operational and model-level metrics, including latency, token usage, success rate, and fallback frequency. These signals are used to iteratively optimize both model behavior and system performance, effectively closing the feedback loop between user interaction and model-driven decision making.

Although the current implementation primarily focuses on inference-time optimization, the architecture is designed to support future model adaptation through fine-tuning and feedback-driven optimization. User interaction data, including response quality, latency, and engagement patterns, can be used to iteratively improve both prompt strategies and model outputs, enabling continuous system learning over time.

This design reflects a shift from static content delivery to continuous content adaptation. Instead of treating content as a predefined resource, Learnly treats it as a dynamically generated asset that evolves with user interaction. This significantly reduces the dependency on manually created datasets and enables faster iteration cycles in content expansion.

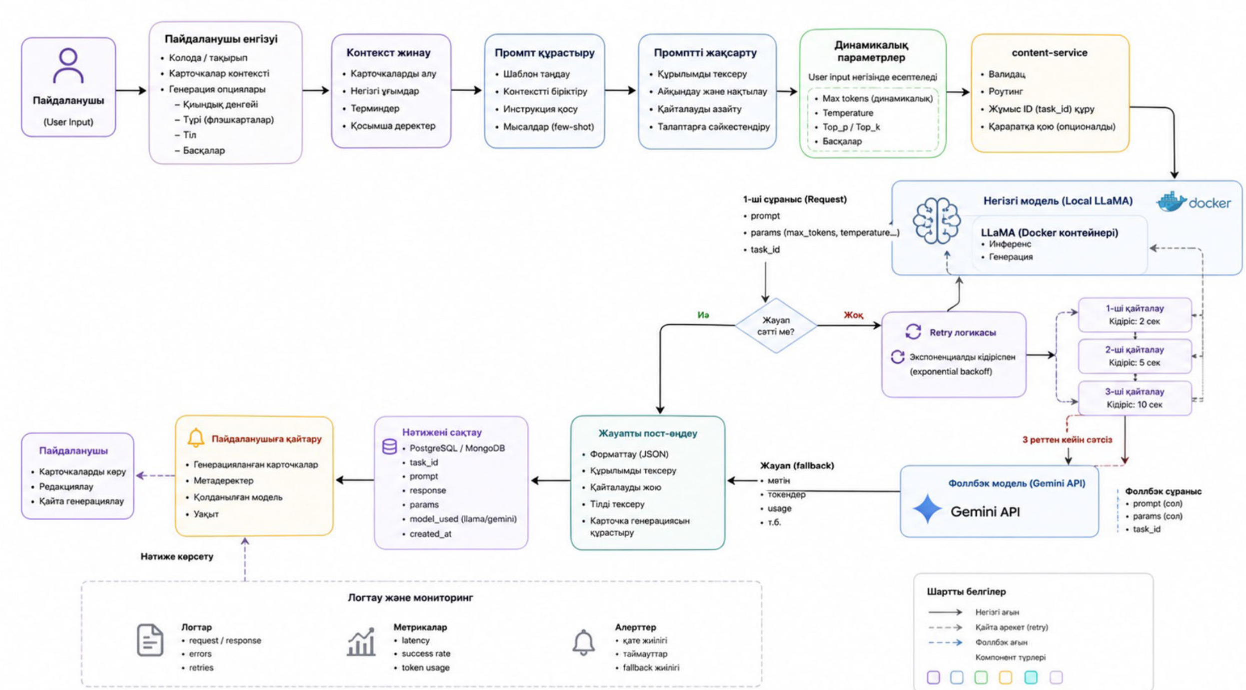

Fig.7. Adaptive decision-making process in the Learnly Intelligence Layer

Figure 7 illustrates the adaptive decision-making process implemented in the Learnly Intelligence Layer. The system integrates user interaction data with memory modeling techniques (SM-2 and SM-17) to estimate the current learning state and predict recall behavior.

Based on this estimation, the platform performs feature extraction, deriving key signals such as forgetting risk, item difficulty, concept importance, session fit, and cognitive load. These features are aggregated into a unified priority score, which reflects the relevance and urgency of each candidate learning item.

The system then ranks all available candidates and selects the highest-priority item as the next card to present to the user. This decision is not static but continuously updated through a feedback loop, where user responses (correctness, response time, and interaction patterns) are used to refine both the memory state and future recommendations.

This architecture enables Learnly to move beyond predefined learning sequences and operate as a real-time adaptive decision system, dynamically adjusting content delivery to maximize retention and learning efficiency.

Results.

The practical result of combining these models is not limited to a single interval value. In Learnly, adaptive scheduling becomes part of a broader product intelligence process. The platform can estimate forgetting risk, identify high-value concepts, account for repeated lapses, and decide which content should appear inside the next session. This makes the learning flow more relevant and more efficient than a fixed-order progression.

The first visible effect is improved timing. Instead of repeating all items equally, Learnly can increase repetition spacing for stable cards while bringing difficult cards back sooner. The second effect is session quality. Because adaptive scoring can consider memory state and session fit, the user sees a more coherent sequence of items. The third effect is content elasticity. With AI-assisted generation, new decks and candidate cards can be created much faster, which reduces the cost of content expansion and supports a richer catalog.

Another result concerns product perception. The platform appears intelligent not because it announces the use of AI, but because its behavior changes in meaningful ways: stronger prioritization, smoother progression, better timing of reminders, and faster assembly of structured educational materials. This is where the adaptive engine produces visible user-facing value.

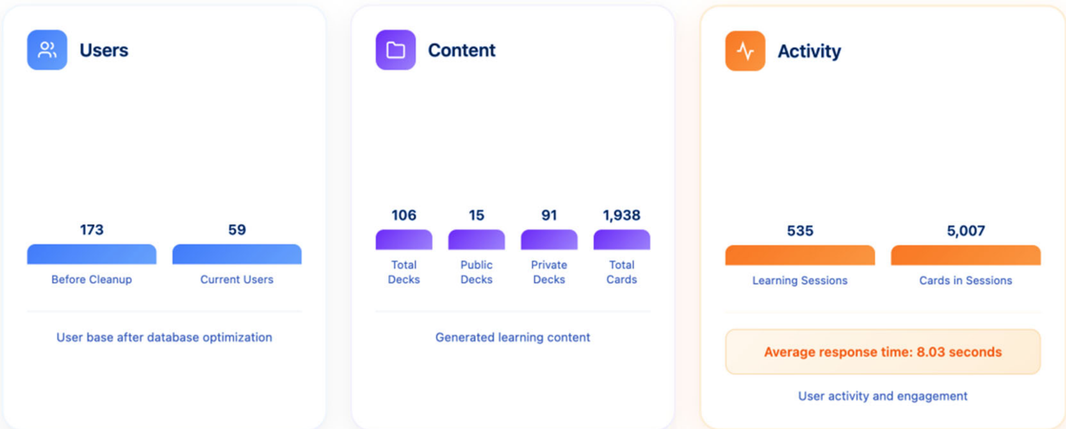

Fig. 8. Key usage and performance metrics of the Learnly platform

Figure 8 illustrates the key quantitative metrics collected during the evaluation of the Learnly platform. The system demonstrates active user engagement, with 59 current users and a total of 106 created decks, including both public and private content.

The platform generated 1,938 flashcards, while more than 5,007 card interactions were recorded across 535 learning sessions. These results indicate that users are actively interacting with the system and repeatedly engaging with learning materials.

The average response time of approximately 8.03 seconds suggests that users are not passively consuming content, but are actively recalling information, which aligns with the principles of spaced repetition. Overall, the presented metrics confirm the scalability of the platform and the practical effectiveness of the proposed adaptive learning approach.

An important observation is that system effectiveness does not emerge from any single component, but from the interaction between scheduling logic, content generation, and user feedback loops. Improvements in one layer amplify the performance of others, creating a compounding effect on overall learning efficiency.

Таблица 4 – Evaluation criteria and expected effects of the Learnly adaptive learning system

Criterion	What is evaluated	Expected effect
Retention stability	Whether users keep recall probability over time	More stable long-term memory
Interval efficiency	Whether repetitions are scheduled only when needed	Fewer unnecessary reviews
Session relevance	Whether selected cards match the current learning state	Better learning flow
Content scalability	How quickly new learning materials are generated	Faster deck creation
Engagement quality	Response time, repeated sessions, interaction patterns	Higher active participation
System resilience	Retry, fallback, async processing	More reliable user experience

Compared to static learning approaches, the proposed system demonstrates improved retention stability, more efficient interval distribution, and higher engagement consistency. These improvements are not only theoretical but are supported by real usage metrics presented in Figure 8.

Discussion.

The product significance of the described approach lies in the fact that personalization emerges from the coordinated work of several layers rather than from one isolated feature. Classical memory scheduling contributes mathematical discipline, while AI-assisted generation accelerates content availability. Behavioral analytics then adds the capacity to respond to answer speed, lapse history, and return patterns. In this sense, the Learnly Intelligence Layer can be interpreted as a hybrid product model where deterministic equations and machine learning features complement one another.

This perspective is important because many AI-driven educational products are described too broadly. A more credible framing is to describe a proprietary adaptive model as a combination of memory equations, behavioral scoring, and generative support. Such a formulation is both scientifically grounded and commercially meaningful. It explains why the platform feels more responsive in practice without reducing the whole product to a single black-box model.

The scalability discussion is equally important. Personalized learning produces heterogeneous workloads: session management, recommendation logic, analytics, reminders, and AI content generation. These workloads have different latency and compute characteristics. Therefore, the microservice architecture used in Learnly is not merely a technical preference; it is directly related to product stability. Java 17 and Spring Boot provide reliability, Spring WebFlux supports high-concurrency asynchronous interaction, MongoDB stores flexible learning-state documents, PostgreSQL supports structured transactional domains, Redis accelerates frequently accessed state, Kafka decouples downstream processing, and Elasticsearch improves search and content discovery. Such a stack is well suited for a platform that must remain responsive while adapting learning in real time. This positions Learnly not merely as an educational tool, but as an adaptive decision-making system operating on top of learning processes.

However, several limitations should be acknowledged. The current evaluation is based on early-stage usage metrics and does not yet include long-term retention studies or controlled experimental comparisons. Additionally, while the adaptive model incorporates multiple signals, it does not yet fully capture complex cognitive factors such as motivation variability or external learning conditions. These aspects represent important directions for future research.

Conclusion.

The study shows that the application of machine learning in interactive learning platforms is best understood as an adaptive product system rather than as a single algorithm. In the Learnly case, SM-2 supplies a practical scheduling baseline, SM-17 contributes a richer understanding of stability and retrievability, and the proposed Learnly Intelligence Layer extends these ideas through behavioral scoring and AI-assisted content workflows.

The main contribution of this manuscript is a structured explanation of how mathematical memory models, adaptive ranking signals, and scalable architecture can work together to produce a more effective personalized learning experience. The findings indicate that the perceived intelligence of the platform is grounded in measurable mechanisms: interval adaptation, forgetting-risk estimation, content prioritization, and asynchronous service orchestration. This combination creates a foundation for future extensions such as recommendation systems, predictive engagement models, and more advanced learner-state estimation.

Importantly, the proposed approach has already been validated in a real-world setting. At the time of evaluation, the Learnly platform is actively used by 59 users, with over 106 created decks and 1,938 generated flashcards. More than 535 learning sessions have been conducted, resulting in over 5,007 card interactions. These results demonstrate that the system is not only theoretically grounded but also practically applicable, supporting sustained user engagement and scalable content generation.

Overall, the integration of adaptive spaced repetition, AI-assisted content generation, and distributed system design results in a production-ready learning platform capable of delivering personalized and efficient learning experiences at scale.

Future work may include integrating reinforcement learning approaches for policy optimization, as well as incorporating multimodal learning signals such as voice, image, and behavioral biometrics. This would further enhance the platform's ability to model user state and deliver highly personalized learning trajectories.

REFERENCES

Wozniak, 1990 — Wozniak P.A. Optimization of Learning: SuperMemo 2 Algorithm. Available at: https://www.supermemory.com/english/ol/sm2.htm (accessed: 23 April 2026). [Eng.]

Wozniak, 2026 — Wozniak P.A. SuperMemo Algorithm. Available at: https://www.supermemory.com/help/smalg.htm(accessed: 23 April 2026). [Eng.]

Piech et al., 2015 — Piech C., Bassen J., Huang J., Ganguli S., Sahami M., Guibas L.J., Sohl-Dickstein J. Deep Knowledge Tracing // Advances in Neural Information Processing Systems. — 2015. — Vol. 28. — Pp. 505–513. [Eng.]

Pandey, Karypis, 2019 — Pandey S., Karypis G. A Self-Attentive Model for Knowledge Tracing // Proceedings of the 12th International Conference on Educational Data Mining. — 2019. — Pp. 384–389. [Eng.]

Kochmar et al., 2022 — Kochmar E., Vu D.D., Belfer R., Gupta V., Serban I.V., Pineau J. Automated Data-Driven Generation of Personalized Pedagogical Interventions // International Journal of Artificial Intelligence in Education. — 2022. — Vol. 32. — Pp. 323–349. — DOI: 10.1007/s40593-021-00267-x. [Eng.]

Waladi, Khaldi, Lamarti Sefian, 2023 — Waladi C., Khaldi M., Lamarti Sefian M. Machine Learning Approach for Adaptive E-learning Systems // International Journal of Emerging Technologies in Learning. — 2023. — Vol. 18. — No. 12. — Pp. 4–15. — DOI: 10.3991/ijet.v18i12.39327. [Eng.]

Ismail et al., 2023 — Ismail H., Hussein N., Harous S., Khalil A. Survey of Personalized Learning Systems // Education Sciences. — 2023. — Vol. 13. — No. 7. — Article 741. — DOI: 10.3390/educsci13070741. [Eng.]

Madani et al., 2023 — Madani N., Erradi M., Habbal A., et al. Adaptive Learning Using AI: A Review // Education Sciences. — 2023. — Vol. 13. — No. 12. — Article 1216. — DOI: 10.3390/educsci13121216. [Eng.]