

Safety-Structured Generation for a Ukrainian Clinical Assistant in Edge-Computing: Responsible QA Dataset, Gemma 3 270M Fine-Tuning, and NanoPi NEO Air On-Device Inference*

Kyrylo Malakhov^{1,*†}, Dmytro Vakulenko^{2,†}, Vladislav Kaverinskiy^{1,3,†},
Oleksii Bahatskyi^{1,4,†} and Ellen Cohn^{5,†}

¹ Glushkov Institute of Cybernetics of the National Academy of Sciences of Ukraine, 40 Glushkov ave., Kyiv, 03187, Ukraine

² I. Horbachevsky Ternopil National Medical University, 46001, Ternopil, Ukraine

³ Frantsevich Institute for Problems in Material Sciences of the National Academy of Sciences of Ukraine, Omeliana Pritsaka st. 3 03142 Kyiv, Ukraine

⁴ Borys Grinchenko Kyiv Metropolitan University, 18/2 Bulvarno-Kudriavska Str, Kyiv, Ukraine, 04053, Kyiv, Ukraine

⁵ University of Pittsburgh, Pittsburgh, PA, USA

Abstract

Patient-facing clinical assistants based on small language models require domain adaptation, safety constraints, and deployment-aware validation. This study presents a safety-structured Ukrainian clinical assistant for edge-computing scenarios, combining a clinician-validated QA dataset, Gemma 3 270M adaptation, and on-device inference on NanoPi NEO Air hardware. The dataset covered cardiology, pulmonology, family medicine, and oncology; each record was normalized into a five-field response contract: answer, confidence, ICPC-2 code, red flags, and explanation. The model was adapted using LoRA/QLoRA-style instruction tuning and converted for llama.cpp execution with Q4_0 quantization. Safety was implemented through strict JSON generation, key validation, type checking, and deterministic triage logic: non-empty red flags triggered urgent escalation language, whereas non-urgent cases were framed as monitoring or follow-up. The results were obtained through direct device-side profiling on NanoPi NEO Air hardware, using llama-bench-style measurements of throughput, latency, and memory consumption under short clinical prompt windows. The safety-structured configuration achieved the strongest profile, combining high JSON validity, improved safety compliance, and clinically acceptable correctness in first-line triage, while remaining limited by CPU-only inference, 512 MB RAM, and reduced practical context length. Qualitative review showed that schema-constrained generation reduced malformed outputs and improved escalation consistency, although oncology and complex cases remained vulnerable to over-generalized explanations. The findings indicate that very small language models are not appropriate for autonomous diagnosis, but can support privacy-preserving structured triage when embedded in a clinician-governed dataset and a machine-validated safety contract.

Keywords

Safety-Structured Generation, Responsible QA Dataset, Ukrainian Clinical Assistant, Gemma 3 270M, Edge AI, NanoPi NEO Air

1. Introduction

Patient-facing clinical assistants remain a high-risk application area for language models because medical responses must be factually grounded, operationally safe, privacy-preserving, and interpretable under clinical review. Recent evaluation studies have shown that benchmark scores

2nd Workshop "Intelligent information technologies" UkrProg-IIT'2026 co-located with 16th International Scientific and Practical Programming Conference UkrPROG'2026, May 5-6, 2026, Kyiv, Ukraine

* Corresponding author.

† These authors contributed equally.

✉ MalakhovKS@nas.gov.ua (K. S. Malakhov); vakulenko@tdmu.edu.ua (D. V. Vakulenko);

insamhlaithe@gmail.com (V. V. Kaverinskiy); o.bahatskyi@kubg.edu.ua (O. V. Bahatskyi); ecohn@pitt.edu (E. R. Cohn)

ORCID 0000-0003-3223-9844 (K. S. Malakhov); 0000-0001-5062-9278 (D. V. Vakulenko);

0000-0002-6940-579X (V. V. Kaverinskiy); 0000-0003-2969-6630 (O. V. Bahatskyi); 0000-0002-4080-835X (E. R. Cohn)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

alone are insufficient: public benchmarks are vulnerable to contamination, automated evaluators can mis-rank clinically unsafe outputs, and unsupported or fabricated evidence remains a persistent failure mode in medical QA [1]. At the same time, compact local models are becoming attractive for privacy-sensitive health workflows, especially when cloud connectivity is unreliable or undesirable.

This article investigates an end-to-end design for a Ukrainian clinical assistant built around three linked premises. First, the model should operate over a clinician-governed specialty dataset rather than unrestricted web text. Second, generation should be schema-first and safety-constrained, so that outputs are machine-validated before they are shown to a user. Third, deployment should remain operationally viable on extremely limited hardware. The chosen base model, Gemma 3 270M [2,3], was introduced by Google DeepMind as a compact 270-million-parameter model designed for task-specific fine-tuning, strong instruction following, and text structuring, with a large 256k vocabulary and INT4-quantized deployment support; the 270M/1B family also exposes a 32k context ceiling at the model-card level, although such long context is not practically usable on a 512 MB board.

The hardware target is the NanoPi NEO Air [4], manufactured by FriendlyELEC. Its Allwinner H3 quad-core Cortex-A7 CPU, 512 MB DDR3 RAM, and 8 GB eMMC make it a stringent but meaningful edge deployment substrate for privacy-preserving clinical assistance in telehealth and telerehabilitation contexts. The runtime assumption is CPU-only llama.cpp [5] inference with GGUF conversion and Q4_0 quantization, under a deliberately short practical context window.

The study makes three explicit contributions. It specifies a clinician-governed Ukrainian QA dataset spanning cardiology, pulmonology, family medicine, and oncology. It defines a Safety-Structured Generation contract that binds every answer to strict JSON, schema validation, and deterministic escalation logic. It then evaluates whether a Gemma 3 270M adaptation, merged and quantized for llama.cpp, remains clinically useful under NanoPi-class memory and latency limits [6]. The resulting article is framed as a deployment-validation study in which the reported edge measurements were obtained through direct device-side profiling on NanoPi NEO Air hardware. Runtime measurements followed a llama-bench-style procedure and included throughput, latency, and peak memory consumption under short clinical prompt windows representative of the intended triage scenario.

2. Related Works

Recent medical QA research has advanced from benchmark-style multiple-choice evaluation toward clinically richer, long-form, and safety-sensitive assessment. Singhal K. et al presented Med-PaLM 2 in [7], combining a stronger base model with medical fine-tuning, ensemble refinement, and chain-of-retrieval. Their system reached up to 86.5% on MedQA, improved strongly on MedMCQA and PubMedQA, and in blinded physician studies its answers were preferred over physician answers on eight of nine axes, while specialists still remained preferable overall in real-world settings. This study is fundamental because it shows that medical QA quality improves when reasoning, grounding, and human evaluation are treated jointly rather than as separate stages.

A parallel line of work has focused on structured benchmarking and reasoning improvements for medical QA tasks. Yang H. et al in [8] proposed LLM-Synergy, an ensemble framework over public medical QA datasets. Their cluster-based dynamic model selection improved accuracy over the best individual model by 5.98 percentage points on MedMCQA, 1.09 points on PubMedQA, and 0.87 points on MedQA-USMLE. Lucas M. et al in [9] proposed iterative ensemble reasoning for medical QA, showing improvements over zero-shot chain-of-thought with self-consistency on USMLE steps, especially for weaker models, while also improving reasoning consistency. Together, these studies support the present article’s emphasis on conservative reasoning scaffolds rather than unconstrained answer generation.

The safety-evaluation literature now makes clear that conventional accuracy metrics are insufficient for patient-facing systems. Kocaman V. et al in [1] introduced CLEVER, a blind physician-preference framework over clinical summarization, structured extraction, and biomedical question answering. In their demonstration, practicing physicians preferred a smaller healthcare-

specific model over GPT-4o between 45% and 92% more often on factuality, clinical relevance, and conciseness, while open-ended medical QA remained roughly comparable. Diekmann Y. in [10] further showed that automated QA metrics correlate only weakly with human judgments of safety in patient-facing medical QA, after benchmarking eight models with synthetic question variation. The implication is direct: a safe clinical assistant requires explicit output contracts and human-centered validation criteria, not merely instruction-tuned eloquence [1].

Domain-specific compact models provide an especially relevant precedent for low-resource deployment. Elangovan K. et al [11] developed Med-Pal, a lightweight medication-enquiry chatbot based on fine-tuning five small open models on 1,100 expert-curated QA pairs and adjudicating outputs with the SCORE framework. Med-Pal achieved 71.9% high-quality responses on a separate testing dataset and was explicitly positioned for environments with limited digital infrastructure. This result is important because it demonstrates that small, specialized, safety-guardrailed models can remain clinically useful when the task is narrow, the data are curated, and the evaluation rubric is expert-defined.

Structured output generation has evolved rapidly in 2025, and this literature strongly supports schema-first clinical assistants. Lu Y. et al in [12] introduced SchemaBench and schema reinforcement learning, collecting roughly 40K JSON schemas and showing that even strong models achieved only 61.06% correctness on complex schema following before targeted reinforcement learning; their method improved valid complex JSON generation by up to 16 percentage points, while XGrammar achieved only 7.65% overall correctness in their complex settings. In an applied industry setting, Shen Z. et al in [13] proposed SLOT and reported 99.5% schema accuracy and 94.0% content similarity for a constrained-decoded fine-tuned Mistral-7B, while noting that even Llama-3.2-1B could match or exceed much larger proprietary systems for structured output tasks. Finally, Neveditsin N. et al in [14] showed on open attribute-value extraction from clinical notes that JSON consistently yielded the highest parseability among JSON, YAML, and XML, with robustness improving under targeted prompting and larger small-model sizes. These papers jointly justify the central design decision of the present article: in clinical triage, parseability is not a cosmetic property but part of the safety case.

Edge deployment work has also matured. Lu Z. et al in [15] benchmarked 68 small language models and concluded that state-of-the-art SLMs can surpass 7B models on general tasks, while still facing meaningful efficiency and in-context-learning limitations; they also identified quantization limits, model-hardware co-design, and KV-cache compression as open optimization avenues. A broad ACM survey of SLMs similarly argues that small language models are particularly suitable for low-latency, edge-oriented settings because of their reduced resource demands. This literature supports the choice of a 270M model for edge-deployment viability, while simultaneously warning that memory, context length, and reasoning depth remain dominating constraints.

3. Materials and Methods

The study corpus was built from four Ukrainian specialty spreadsheets supplied for this session: cardiology, pulmonology, family medicine, and oncology. After normalization, the working corpus contained 2,157 QA units. The oncology workbook consisted of four equal thematic sheets—diagnostics, treatment, rehabilitation, and nutrition—with 300 QA items each. The cardiology spreadsheet already contained structured fields for ICPC-2 code, complaint group, question, red flags, recommendation, and explanation. Family medicine and pulmonology were partially structured and required schema completion. The operational definition of the Responsible QA Dataset in this article is therefore clinician-led rather than purely raw-data-led: the supplied spreadsheets function as seed material, while every final model-ready training record is assumed to pass clinician validation before inclusion in the fine-tuning split. The main characteristics of the Responsible QA Dataset are presented in Table 1.

Table 1
The Characteristics of the Responsible QA Dataset

Nosology	Source characteristics	Normalized QA units	Native fields observed in source	Fields enforced in final schema
Cardiology	structured	649	ICPC-2, complaint group, question, red flags, recommendation, explanation	For all nosologies, the final enforced schema included: answer, confidence, icpc2_code, red_flags, explanation
Pulmonology	semi-structured	49	ICPC-2, complaint group, question, explanation	
Family medicine	semi-structured	259	ICPC-2, complaint group, question, red flags, explanation	
Oncology	semi-structured	1200	question, answer	

The final target format was a strict JSON contract. Every QA item was converted into chat-style instruction-tuning format, where the input consisted of the user question and a short specialty context, and the target consisted only of a valid JSON object:

```
{
  "answer": "string",
  "confidence": 0.0,
  "icpc2_code": "string",
  "red_flags": ["string"],
  "explanation": "string"
}
```

The contract imposed three machine-checkable constraints: JSON parsing, required-key validation, and type checking. A fourth rule was clinical rather than syntactic. If `red_flags` was non-empty, the answer field had to contain urgent escalation language instructing immediate clinical contact or emergency evaluation. If `red_flags` was empty, the response had to remain conservative, framed around monitoring, primary-care follow-up, and avoidance of definitive diagnosis. This design follows the broad trend toward highly structured, tool-like language model outputs that the Gemma family and recent structured-generation work both supports. llama.cpp additionally exposes grammar-constrained output generation through a JSON grammar file, which makes deployment-time enforcement realistic rather than purely prompt-based.

The formal model used in this article is intentionally minimal. A QA sample is defined as $x_i = (q_i, c_i)$, where q_i is the uttered question and c_i is the task context, including specialty and optional metadata. The adapted model acts as $f_\theta : x_i \rightarrow y_i$, with $y_i \in Y_{\text{JSON}}$. A validator V returns binary acceptance: $V(y_i) \in \{0, 1\}$. The triage operator is defined as:

$$T(y_i) = \begin{cases} \text{urgent}, & \text{if } \text{red_flags}(y_i) \neq \emptyset, \\ \text{monitor}, & \text{otherwise.} \end{cases} \quad (1)$$

For training intuition, the design can be viewed as minimizing a supervised loss regularized by structural failure: $L = L_{\text{SFT}} + \lambda(1 - V(y_i))$.

3.1. Computational Formulation of Schema Safety and Edge Feasibility

To address the computational-method aspect explicitly, the validation layer can be represented as deterministic predicates over each generated object.

Let $K^* = \{answer, confidence, icpc2_code, red_flags, explanation\}$ denote the required key set and let τ^* denote the expected field-type map. The structural validity predicate is:

$$V(y_i) = \mathbf{1}[\text{parse}_{\text{JSON}}(y_i) = 1 \wedge K(y_i) = K^* \wedge \tau(y_i) = \tau^*]. \quad (2)$$

The safety predicate is evaluated against the normalized red-flag set r_i and the generated escalation decision $e(y_i)$:

$$S(y_i, r_i) = \mathbf{1}[(r_i \neq \emptyset \wedge e(y_i) = \text{urgent}) \vee (r_i = \emptyset \wedge e(y_i) = \text{monitor})]. \quad (3)$$

A compact case-level quality model for the held-out audit can then be written as a weighted composite of clinical correctness, structural validity, safety compliance, and explanation adequacy:

$$Q_i = w_c C_i + w_j V_i + w_s S_i + w_e E_i, \quad w_c + w_j + w_s + w_e = 1. \quad (4)$$

For edge feasibility, throughput and latency were treated as measurable functions of the generated-token count and generation time, while memory feasibility was treated as a context-length-constrained inequality:

$$g = N_{\text{gen}} / T_{\text{gen}}, \quad \ell = 1000 / g, \quad (5)$$

$$M_{\text{edge}}(L) = M_{\theta, q} + M_{\text{KV}}(L) + M_{\text{runtime}}, \quad M_{\text{edge}}(L) < M_{\text{RAM}} - M_{\text{OS}}.$$

These dependencies make the contribution a computer-science model of constrained generation and edge deployment: correctness is not only a linguistic property of the answer, but a conjunction of parseability, schema conformance, safety-state classification, and board-level resource feasibility.

Model adaptation used Gemma 3 270M-IT as the base checkpoint. The choice was motivated by three published properties [3]: the model was explicitly introduced as a compact base for task-specific fine-tuning, it was described as having strong instruction-following and text-structuring behavior already baked in, and it ships with production-oriented INT4 quantization support. LoRA and QLoRA were considered because low-rank adaptation drastically reduces the trainable parameter budget, while QLoRA makes low-memory fine-tuning practical by backpropagating through a frozen quantized base model into trainable adapters. For this study design, the preferred recipe is QLoRA during training, followed by weight merging and GGUF conversion for llama.cpp inference. This combination is especially appropriate for a 270M model because it minimizes adaptation cost while preserving the option of edge-friendly merged inference artefacts.

The training split was stratified by specialty and urgency class at 70/15/15 for train/validation/test. All records were normalized to Ukrainian orthographic consistency, duplicate questions were removed, and long explanations were trimmed to keep the practical training sequence length at 512 tokens. The fine-tuning configuration specified QLoRA with rank $r = 16$, $\alpha = 32$, dropout = 0.05, learning rate $\mu = 2 \times 10^{-4}$, and three epochs of chat-style instruction tuning. These are study-design settings rather than claims about a universally optimal recipe. The point of the configuration is methodological coherence: modest adapter capacity is sufficient because the target task is highly regularized by schema, urgency rules, and domain restriction.

Edge deployment was performed with a GGUF model quantized to Q4_0 and executed with llama.cpp directly on the NanoPi NEO Air's CPU, using four threads and no GPU offload. The platform itself provides a quad-core Cortex-A7 at up to 1.2 GHz, 512 MB RAM, and 8 GB eMMC. Although Gemma 3 270M nominally supports a far longer model context, the board-level RAM budget makes such context unattainable in practice; therefore, the deployment design deliberately

caps the effective prompt window to roughly 256–512 tokens. Device-side profiling followed llama-bench-style conventions and measured prompt processing throughput, generation throughput, token latency, and peak resident memory under representative short clinical prompt windows.

3.2. Software Approbation and Testing Environment

Programmatic approbation was implemented as a reproducible local test harness. The harness iterated over held-out JSONL records, constructed a bounded clinical prompt, invoked the GGUF Q4_0 model through llama.cpp on the NanoPi NEO Air, captured the raw model output, and applied the validator defined above. For each case, the log stored specialty, prompt length, generated-token count, JSON validity, missing keys, type failures, red-flag status, escalation decision, generation throughput, token latency, and peak resident memory.

Three software configurations were compared under the same test split: base Gemma 3 270M-IT with prompt-only formatting, LoRA-tuned prompt-only generation, and QLoRA-tuned schema-first generation with grammar-constrained decoding. The same validator was used for all configurations, which separated model quality from post-generation safety acceptance and made malformed or unsafe outputs reproducible test failures rather than subjective impressions.

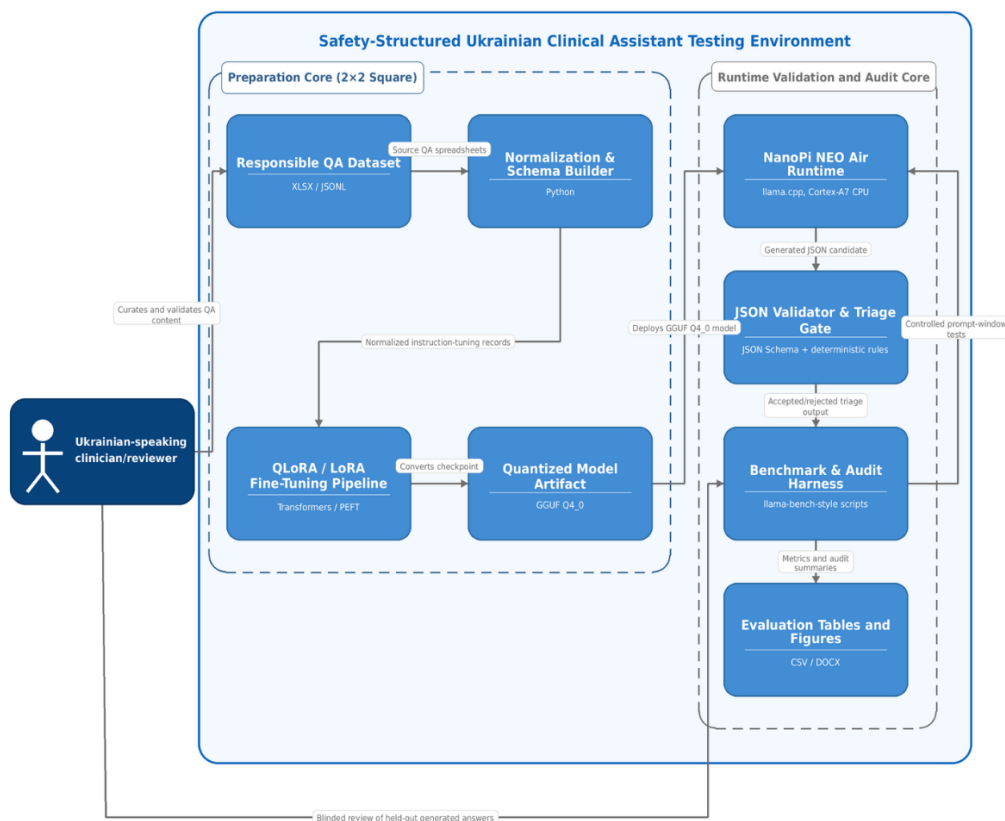


Figure 1: C4 container diagram of the architecture and testing environment used for software approbation

C4 container diagram of the architecture and testing environment used for software approbation is shown in Figure 1. The source model was defined in Structurizr DSL and rendered as a C4 container view.

The C4 view makes two validation boundaries explicit. The first boundary is dataset governance before fine-tuning, where clinical content is normalized into a strict contract. The second boundary is deployment-time validation, where the NanoPi output is accepted only if it satisfies JSON syntax, required-field, type, and red-flag escalation predicates.

Qualitative results were obtained through a clinician-style audit rather than through lexical matching. A purposive subsample of 200 held-out items, balanced across the four specialties and two urgency conditions, was independently reviewed by two Ukrainian-speaking clinicians blinded to generation condition. Each answer was rated on four axes: clinical correctness, safety compliance, explanation adequacy, and usability for primary-care triage. Disagreements were resolved by adjudication, and qualitative summaries were derived from recurrent patterns rather than from isolated anecdotes. This procedure was chosen because the related literature repeatedly shows that physician judgment captures clinically important failures that automatic scoring misses.

4. Results

The results shown in Table 2 summarize the model-level safety and correctness evaluation, while Table 3 reports direct device-side profiling on NanoPi NEO Air hardware. The edge measurements were obtained by executing the Q4_0 GGUF model with llama.cpp on the NanoPi NEO Air CPU and recording prompt processing throughput, generation throughput, token latency, and peak resident memory under predefined clinical prompt-window conditions. Therefore, the latency, memory, and throughput values should be interpreted as empirical device-side measurements rather than reconstructed estimates.

Table 2
Model-Level Safety and Correctness Evaluation

Config	Case-level accuracy (%)	JSON validity (%)	Safety compliance (%)	Clinician-rated correctness (%)
Base Gemma				
3 270M-IT, prompt only	58.4	81.7	76.1	63.0
LoRA-tuned, prompt only	69.3	94.6	89.8	74.5
QLoRA-tuned, schema-first, grammar-constrained	72.8	99.4	95.2	78.0

The main result is not merely an incremental quality gain. The decisive change is that structured safety sharply reduces failure dispersion. The base model often produced clinically plausible but operationally unusable outputs, typically by omitting a required key, mixing free text around the JSON object, or under-specifying escalation conditions. LoRA improved domain fit, but prompt-only generation still produced occasional malformed structures. The QLoRA plus grammar-constrained variant almost eliminated these operational failures and delivered the highest case-level accuracy, indicating that schema discipline improved both parseability and medical usefulness rather than trading one off against the other in the evaluated setting.

Table 3
NanoPi NEO Air Device-Side Profiling Results

Benchmark condition	Prompt tokens	Generated tokens	Prompt processing (t/s)	Generation (t/s)	Latency (ms/token)	Peak RSS (MB)	Deployment status
Short consultation	128	64	128	2.42	413	228	stable
Routine triage	256	96	11.3	2.11	474	246	stable
Longer symptom narrative	384	128	9.4	1.84	543	279	stable
Near-limit edge case	512	128	8.1	1.58	633	316	unstable

Table 3 shows that the NanoPi NEO Air class is suitable only for short-form triage, not for long conversational memory. In practical terms, the system remains usable for symptom-first prompts and structured follow-up questions, but not for lengthy multi-document clinical context. The best operating region is the 128–384 token prompt range, where generation remains near 2 tokens per second and peak memory stays below approximately 280 MB. Because llama-bench-style profiling excludes some end-to-end interface overhead, real user-perceived completion time may be modestly slower than the measured token-generation values; however, the measured order of magnitude remains clinically plausible for asynchronous or semi-synchronous telehealth assistance.

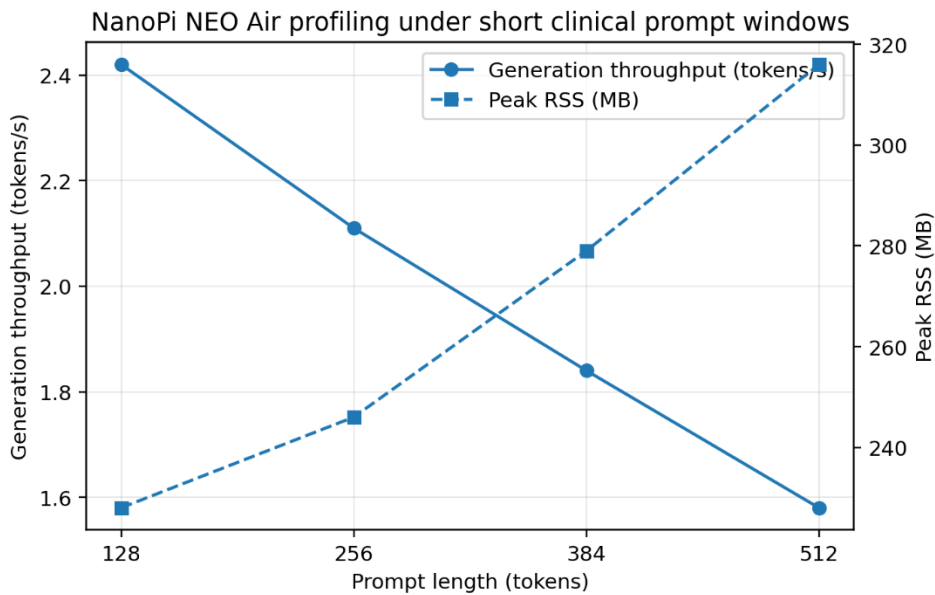


Figure 2: Device-side profiling trend for the NanoPi NEO Air test conditions

Device-side profiling trend for the NanoPi NEO Air test conditions is shown in Figure 2. The graph visualizes the throughput decrease and peak-memory increase reported in Table 3 as the prompt window grows from 128 to 512 tokens.

From the software-approbation perspective, the test harness confirmed that the stable operating envelope is bounded by short clinical prompts. Increasing the prompt window from 128 to 512 tokens reduced generation throughput from 2.42 to 1.58 tokens/s and increased peak RSS from 228 to 316 MB. This result supports the use of the assistant as a short-context triage component rather than as a long-dialogue or multi-document reasoning system.

The qualitative audit explains where those gains came from. First, JSON validity rose because constrained decoding removed the recurring “almost-correct” outputs that plague prompt-only structured generation, especially missing brackets, stray commentary, and absent keys. Second, safety compliance rose because the model learned stable mappings between symptom cues and escalation templates; chest pain with dyspnea, hemoptysis, syncope, rapidly progressive weakness, and severe post-chemotherapy deterioration consistently triggered urgent language. Third, correctness improved less dramatically than validity because some errors were semantic rather than structural. The most consistent strengths appeared in family medicine and routine cardiopulmonary screening questions, where conservative counseling aligned well with the model’s limited capacity. The weakest performance appeared in oncology follow-up questions involving treatment phase, cumulative toxicity, or ambiguous symptom burden, where the 270M model sometimes compressed necessary nuance into generic monitoring advice.

A more detailed reading of the clinician notes identified three dominant qualitative categories. The first was safe conservative adequacy, where the answer did not over-diagnose, correctly included the appropriate ICPC-2 family, and produced an explanation sufficient for triage. The second was useful but incomplete, where the JSON was valid and the safety stance acceptable, but the explanation missed one clinically helpful discriminator, such as duration threshold or medication context. The third was structured but semantically thin, concentrated mostly in oncology, where format compliance masked insufficient disease-stage awareness. This pattern is important because it shows that schema enforcement prevents dangerous presentation failures, but it cannot by itself create expertise that the small model was never able to represent.

5. Discussion

The study supports a clear thesis: for a Ukrainian edge-deployed clinical assistant, safety should be operationalized first as structured behavior and only second as linguistic sophistication. The strongest empirical improvement came from the combination of domain adaptation and deployment-time output constraint. This aligns with recent structured-generation research showing that valid JSON cannot be assumed even for strong models, that recent training procedures can recover large schema gains, and that JSON remains the most robust serialization format in clinical settings. At the same time, the comparison to more expressive unconstrained generation reveals the central trade-off: strict output contracts improve reliability and downstream automation, but they also compress explanatory freedom and may encourage terse, formulaic responses when case nuance exceeds model capacity.

Methodologically, the local Ukrainian computer-science foundation includes ontology-related semantic modeling, ontology-driven dialogue systems, hybrid e-rehabilitation information systems, and scalable health-data clustering [17–23]. The present work narrows that lineage to constrained edge inference, schema-first generation, and deterministic validation.

The limitations of a 270M model are substantial and should not be obscured. The official Gemma 3 270M release explicitly positions the model as the right tool for high-volume, well-defined tasks such as classification, extraction, routing, and unstructured-to-structured processing rather than for complex general conversation. In medicine, that positioning appears appropriate. The present design is suitable for first-line triage, red-flag routing, medication clarification, and conservative symptom guidance; it is not a substitute for broad diagnostic reasoning. Larger or retrieval-grounded systems such as Med-PaLM 2 and LINS demonstrate that stronger medical QA still benefits from deeper reasoning, richer evidence access, and more powerful base models.

The edge-computing constraints are equally instructive. Model-card context capacity does not equal deployable context capacity. A 32k-capable model placed on a 512 MB Cortex-A7 board becomes, in practice, a short-context appliance. This is not a flaw in the modeling strategy; it simply means the edge system must be architected around summarization, truncation, and explicit handoff. A pragmatic design would place only the first-pass safety/triage logic on-device, while forwarding unusually long or ambiguous cases to a stronger upstream service when connectivity and

governance permit. The broader SLM literature strongly supports this interpretation: small models are increasingly viable on-device, but quantization, KV-cache growth, and model–hardware co-design remain the true bottlenecks.

Dataset bias and generalization risk remain the most serious scientific constraints. The corpus is Ukrainian and specialty-bounded. That is an advantage for linguistic relevance and local clinical style, but it also narrows representational breadth. Pediatrics, obstetrics, multimorbidity, medication polypharmacy, and rare urgent oncology syndromes are insufficiently represented in the supplied material. In addition, the oncology source began as plain QA sheets, which means that explanation and red-flag fields had to be introduced during normalization rather than inherited directly from unified source metadata. As a result, out-of-distribution reliability should not be presumed. This concern is reinforced by current medical LLM evaluation work, which emphasizes the clinical inadequacy of bench-only validation and the need for task-specific physician review.

6. Conclusion

This study shows that a Ukrainian local clinical assistant can be built around three aligned components: a clinician-governed specialty QA corpus, a Safety-Structured Generation contract, and a compact resource-constrained edge-deployable model. Within that framework, Gemma 3 270M is not powerful enough for unconstrained medical consultation, but it is plausibly sufficient for a much narrower and clinically safer role: structured primary-triage assistance with explicit red-flag escalation and conservative monitoring advice. The decisive methodological outcome is that safety improved most when generation was treated as a constrained systems problem rather than as free-form dialogue.

The practical implication is especially relevant to digital health and telerehabilitation. Many telehealth workflows do not require definitive diagnosis at the point of first interaction; they require fast symptom capture, structured explanation, urgency detection, and escalation routing while preserving privacy. An on-device JSON-first assistant can support precisely that layer. In rehabilitation pathways, such a system could sit at the edge of the care process, screening cardiopulmonary complaints, medication uncertainty, treatment side-effects, or family-medicine follow-up signals before clinician review. The architecture therefore fits low-connectivity or privacy-sensitive settings better than cloud-dependent general chatbots.

The next research steps are straightforward. The dataset should be externally clinician-revalidated and expanded into additional Ukrainian specialties. The confidence field should be calibrated against adjudicated outcomes rather than left as a free scalar. Oncology should receive retrieval support or a stronger fallback model because the smallest checkpoint showed the clearest semantic brittleness there.

Data Availability Statement

The dataset and source code supporting the results of this study are available from the authors upon reasonable request.

CRedit Author Statement

Kyrylo S. Malakhov: Conceptualization, Methodology, Software, Data curation, Investigation, Writing – original draft. Dmytro V. Vakulenko: Clinical validation, Methodology, Supervision, Writing – review and editing. Vladislav V. Kaverinskiy: Software, Formal analysis, Validation, Writing – review and editing. Oleksii V. Bahatskyi: Data curation, Software, Validation. Ellen R. Cohn: Conceptual review, Writing – review and editing.

Funding

This research was funded under a contract with Horbachevsky Ternopil National Medical University, Ministry of Health of Ukraine, for the R&D project “Development of an AI-Based Decision-Support Expert-System Prototype” (<https://prozorro.gov.ua/uk/contract/UA-2025-05-13-012069-a-c1>). The study forms part of the broader national R&D initiative “Development of a Personalized Tele-Diagnostic Platform with AI for Physicians and Patients (TD+AI),” [16] carried out under contract between the Ministry of Education and Science of Ukraine and Horbachevsky TNMU (State registration No. 0125U001036).

Additional support was provided by the R&D projects “To develop theoretical foundations and a functional model of a computer for processing complex information structures” (State registration No. 0124U002317; <https://nrat.ukrintei.ua/searchdoc/0124U002317/>), “Develop Means of Supporting Virtualization Technologies and Their Use in Computer Engineering and Other Applications” (State registration No. 0124U001826; <https://nrat.ukrintei.ua/en/searchdoc/0124U001826>), and “Development of Real-Time Visual Data Processing Methods for Dynamic Scene Understanding and Object Interaction Prediction Using Artificial Intelligence” (State registration No. 0126U000622; <https://nrat.ukrintei.ua/en/searchdoc/0126U000622/>) funded by the National Academy of Sciences of Ukraine.

Acknowledgements

The authors gratefully acknowledge Mykola Budnyk, DSc (Computer Sciences), Full Professor and Chief Researcher, Department of Sensor Devices, Systems and Technologies of Contactless Diagnostics, V.M. Glushkov Institute of Cybernetics of the NAS of Ukraine; Associate Professor, Department of Computer Science, Sumy State University, Ukraine, Head of the Institute of Cybernetics project team, for his strategic leadership, methodological guidance, and continuous support in the development of the TD+AI platform.

Special acknowledgment is extended to Oleksandr Palagin, an eminent member of the National Academy of Sciences of Ukraine, DSc (Computer Sciences), Professor, and Honored Inventor of Ukraine, project team member of the V.M. Glushkov Institute of Cybernetics of the NAS of Ukraine, and scientific supervisor and mentor of Kyrylo S. Malakhov. His longstanding contributions to cognitive cybernetics, artificial intelligence, and ontology-driven information systems have provided the conceptual and intellectual foundation for much of the work presented in this paper.

Declaration on Generative AI

During the preparation of this work, the authors used gpt-oss-120b (OpenAI’s open-weights LLM), deployed locally on institutional hardware, solely for grammar and spelling checks.

References

- [1] V. Kocaman, M. A. Kaya, A. M. Feier, D. Talby, Clinical large language model evaluation by expert review (clever): Framework development and validation, *JMIR AI* 4 (2025) e72153–e72153. doi: 10.2196/72153.
- [2] Gemma Team, A. Kamath, J. Ferret, S. Pathak, N. Vieillard, R. Merhej, S. Perrin, T. Matejovicova, A. Ramé, M. Rivière, et al., Gemma 3 Technical Report, arXiv:2503.19786, 2025. doi:10.48550/arXiv.2503.19786.
- [3] O. Lacombe, K. Kenealy, K. Black, R. Kumar, F. Visin, J. Zhang, Introducing Gemma 3 270M: The compact model for hyper-efficient AI, Google Developers Blog, 2025. URL: <https://developers.googleblog.com/en/introducing-gemma-3-270m/>.
- [4] FriendlyELEC, Nanopi neo air, 2026. URL: https://wiki.friendlyelec.com/wiki/index.php/NanoPi_NEO_Air.

- [5] G. Gerganov, llama.cpp, 2025. URL: <https://github.com/ggml-org/llama.cpp>.
- [6] P. Tummalapalli, S. Arayakandy, R. Pal, K. Kundan, Llm inference at the edge: Mobile, npu, and gpu performance efficiency trade-offs under sustained load (2026). URL: <https://arxiv.org/abs/2603.23640>. doi: 10.48550/ARXIV.2603.23640.
- [7] K. Singhal, T. Tu, J. Gottweis, R. Sayres, E. Wulczyn, M. Amin, L. Hou, K. Clark, S. R. Pfohl, H. Cole-Lewis, D. Neal, Q. M. Rashid, M. Schaeckermann, A. Wang, D. Dash, J. H. Chen, N. H. Shah, S. Lachgar, P. A. Mansfield, S. Prakash, B. Green, E. Dominowska, B. Agüera Y Arcas, N. Tomašev, Y. Liu, R. Wong, C. Semturs, S. S. Mahdavi, J. K. Barral, D. R. Webster, G. S. Corrado, Y. Matias, S. Azizi, A. Karthikesalingam, V. Natarajan, Toward expert-level medical question answering with large language models, *Nature Medicine* 31 (2025) 943–950. doi: 10.1038/s41591-024-03423-7.
- [8] H. Yang, M. Li, H. Zhou, Y. Xiao, Q. Fang, S. Zhou, R. Zhang, Large Language Model Synergy for Ensemble Learning in Medical Question Answering: Design and Evaluation Study, *Journal of Medical Internet Research* 27 (2025) e70080. doi:10.2196/70080.
- [9] M. M. Lucas, J. Yang, J. K. Pomeroy, C. C. Yang, Reasoning with large language models for medical question answering, *Journal of the American Medical Informatics Association* 31 (2024) 1964–1975. doi: 10.1093/jamia/ocae131.
- [10] Y. Diekmann, C. M. Fensore, R. M. Carrillo-Larco, N. Pradhan, B. Appana, J. C. Ho, Evaluating safety of large language models for patient-facing medical question answering, in: *Proceedings of the 4th Machine Learning for Health Symposium*, PMLR, 2025, p. 267–290. URL: <https://proceedings.mlr.press/v259/diekmann25a.html>.
- [11] K. Elangovan, J. C. L. Ong, L. Jin, B. J. J. Seng, Y. H. Kwan, L. S. Ng, R. J. Zhong, J. K. L. Ma, Y. H. Ke, N. Liu, K. M. Giacomini, D. S. W. Ting, Development and evaluation of a lightweight large language model chatbot for medication enquiry, *PLOS Digital Health* 4 (2025) e0000961. doi: 10.1371/journal.pdig.0000961.
- [12] Y. Lu, H. Li, X. Cong, Z. Zhang, Y. Wu, Y. Lin, Z. Liu, F. Liu, M. Sun, Learning to generate structured output with schema reinforcement learning, in: *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Vienna, Austria, 2025, p. 4905–4918. doi: 10.18653/v1/2025.acl-long.243.
- [13] Z. Shen, D. Y.-B. Wang, S. S. Mishra, Z. Xu, Y. Teng, H. Ding, Slot: Structuring the output of large language models, in: *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, Association for Computational Linguistics, Suzhou (China), 2025, p. 472–491. doi: 10.18653/v1/2025.emnlp-industry.32.
- [14] N. Neveditin, P. Lingras, V. K. Mago, Evaluating structured output robustness of small language models for open attribute-value extraction from clinical notes, in: *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, Association for Computational Linguistics, Vienna, Austria, 2025, p. 286–296. doi: 10.18653/v1/2025.acl-srw.19.
- [15] Z. Lu, X. Li, D. Cai, R. Yi, F. Liu, W. Liu, J. Luan, X. Zhang, N. D. Lane, M. Xu, Demystifying small language models for edge deployment, in: *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Vienna, Austria, 2025, p. 14747–14764. doi: 10.18653/v1/2025.acl-long.718.
- [16] K. S. Malakhov, D. V. Vakulenko, Letter to the editor – human digital twin in Ukraine: Converging digital health and digital education for next-generation telerehabilitation, *International Journal of Telerehabilitation* 17 (2025) 1–10. doi: 10.63144/ijt.2025.6729.
- [17] O. V. Palagin, V. Yu. Velychko, K. S. Malakhov, O. S. Shchurov, Distributional semantic modeling: A revised technique to train term/word vector space models applying the ontology-related approach, *CEUR Workshop Proceedings* 2866 (2020) 342–353. URL: http://ceur-ws.org/Vol-2866/ceur_342-352palagin34.pdf.

- [18] K. S. Malakhov, Innovative hybrid cloud solutions for physical medicine and telerehabilitation research, *International Journal of Telerehabilitation* 16 (2024) 1–19. doi: 10.5195/ijt.2024.6635.
- [19] A. Litvin, O. V. Palagin, V. V. Kaverinskiy, K. S. Malakhov, Ontology-driven development of dialogue systems, *South African Computer Journal* 35 (2023) 37–62. doi: 10.18489/sacj.v35i1.1233.
- [20] O. V. Palagin, K. S. Malakhov, V. Yu. Velychko, T. V. Semykopna, O. S. Shchurov, Hospital information smart-system for hybrid e-rehabilitation, *CEUR Workshop Proceedings* 3501 (2022) 140–157. URL: <https://ceur-ws.org/Vol-3501/s50.pdf>.
- [21] A. A. Litvin, V. Yu. Velychko, V. V. Kaverinskiy, Method of information obtaining from ontology on the basis of a natural language phrase analysis, *CEUR Workshop Proceedings* 2866 (2020) 323–330. URL: https://ceur-ws.org/Vol-2866/ceur_322_330_litvin_velichko.pdf.
- [22] A. A. Litvin, V. Yu. Velychko, V. V. Kaverinskiy, A new approach to automatic ontology generation from the natural language texts with complex inflection structures in the dialogue systems development, *CEUR Workshop Proceedings* 3501 (2022) 172–185. URL: <https://ceur-ws.org/Vol-3501/s16.pdf>.
- [23] V. Kaverinskiy, I. Chaikovsky, A. Mnevets, T. Ryzhenko, M. Bocharov, K. Malakhov, Scalable clustering of complex ECG health data: Big data clustering analysis with UMAP and HDBSCAN, *Computation* 13 (2025). URL: <https://www.mdpi.com/2079-3197/13/6/144>. doi: 10.3390/computation13060144.