Date of publication DEC-31, 2025, date of current version OCT-29, 2025.
www.computingonline.net / computing@computingonline.net

Print ISSN 1727-6209
Online ISSN 2312-5381
DOI 10.47839/ijc.24.4.4349

Empowering Smaller Models: Tuning LLaMA and Gemma with Chain-of-Thought for Ukrainian Exam Tasks

MYKYTA SYROMIATNIKOV[1], VICTORIA RUVINSKAYA[1], NATALIIA KOMLEVA[1]
[1]Odesa Polytechnic National University. Odesa 65044 Ukraine

Corresponding author: Mykyta Syromiatnikov (e-mail: nik.syromyatnikov@gmail.com).

ABSTRACT Leading large language models have demonstrated impressive capabilities in reasoning-intensive tasks, such as standardized educational testing. However, they often require extensive training in low-resource settings with inaccessible infrastructure. Small or compact models, though more efficient, frequently lack sufficient support for underrepresented languages, leaving a performance gap in critical domains. This work explores the potential of parameter-efficient fine-tuning of compact open-weight language models to handle reasoning-intensive tasks in the underrepresented Ukrainian language, building on the findings of the ZNO-Eval benchmark. Parameter-efficient fine-tuning of LLaMA 3.1 (8 billion parameters), LLaMA 3.2 (3 billion parameters), and Gemma 2 (9 billion parameters) models on chain-of-thought solutions resulted in a modest test score improvement of up to 17.4% on complex matching tasks and 1.6% overall compared to tuning on answer letters alone, offering enhanced interpretability and robustness. In addition, the proposed tuning method with joint task topic and step-by-step solution generation outperforms standard chain-of-thought tuning in matching tasks and provides a 5.4% gain over the best LLaMA 3.2 model due to guiding the model to recall and apply domain-relevant information. Contrasting obtained results with zero-shot evaluations of leading open-weight and proprietary models such as Qwen, DeepSeek R1, OpenAI o1 and o3, Gemini, and Claude, highlights that fine-tuning LLaMA and Gemma models with 2,032 step-by-step solutions and 20 to 50 million trainable parameters on a single A100 GPU lets them outperform GPT-4o mini, Mistral Large, and larger open-weight models. This research also evaluates how merging the quantized adapter with the base model influences the generation quality.

KEYWORDS LLM; LLaMA; Gemma; PEFT; Chain-of-Thought, fine-tuning, reasoning, Ukrainian, information technology.

I. INTRODUCTION

In recent years, Large Language Models (LLMs) have demonstrated remarkable proficiency in language understanding tasks, surpassing human-level performance on multiple benchmarks with narrow text understanding tasks, including traditional GLUE [1] and SQuAD [2], as well as challenging reasoning benchmarks like the Massive Multitask Language Understanding (MMLU) benchmark, which encompasses a wide array of subjects and requires advanced reasoning skills [3]. Moreover, in addition to unprecedented accuracy in complex linguistic challenges, giant models like PaLM empower new discoveries in mathematical sciences [4]. At the same time, multimodal solutions like GPT-4o or LLaMA excel in integrating visual and textual information, enabling sophisticated image captioning and visual question-answering [5].

With the advent of the Transformer architecture [6], enhanced attention mechanisms [7], and neural scaling laws [8], language models have revolutionized a wide range of fields, including search, recommendation systems, real-time coding assistance, and even robotics [9], fundamentally reshaping how information is processed and utilized across domains and industries.

Technologies like these serve as a foundation for developing information systems that can be implemented across various domains and integrated with other neural network architectures and machine learning methods. This approach enables the solution of complex, semi-formalized practical tasks that require flexible adaptation and the combination of different intelligent methods [10].

Despite their impressive achievements, LLMs often demonstrate limited capabilities in underrepresented languages. Studies highlight that these models, predominantly trained on high-resource languages like English, struggle to generalize effectively to low-resource languages, resulting in degraded accuracy and robustness [11]. In particular, the ZNO-

Eval benchmark based on standardized exams for Ukrainian school graduates showcased zero-shot strength in factual recall and general knowledge across all models in the National multi-subject test (NMT) subsets for history and geography [12]. However, answer quality drops substantially when dealing with intricate language and specialized terminology of Ukrainian language exam tasks (Table 1). Moreover, the substantial computational resources required to train and deploy large language models introduce additional complexity. The enormous model sizes, exceeding tens or hundreds of billions of parameters, require extensive hardware capabilities, which makes them less suitable for organizations and individuals with limited resources. This scalability issue highlights the importance of more efficient, compact models that deliver relatively high performance without the associated computational overhead.

Table 1. Sample tasks from the Ukrainian language exam along with their English translations

Task description	English translation of task description
Завдання з вибором однієї правильної відповіді: Суфікс **-ин-** має однакове значення в усіх словах, ОКРІМ А - соломина Б - бадилина В - височина Г - стеблина	Single correct answer task: The suffix **-ин-** has the same meaning in all of the following words EXCEPT A – соломина (straw piece) B – бадилина (leafy-stalk piece) C – височина (height / highland) D – стеблина (stem piece)
Завдання на встановлення відповідності (логічні пари): З'ясуйте, якими частинами мови є виділені слова в реченні (цифра позначає наступне слово). Сучасна людина, щоб бути (1)**успішною**, має вчитися (2)**впродовж** (3)**усього** життя, (4)**опановуючи** нові галузі знань. А - займенник Б - прикметник В - форма дієслова (дієприкметник) Г - форма дієслова (дієприслівник) Д - прийменник	Matching task (logical pairs): Determine which parts of speech the highlighted words are in the following sentence (the number indicates the word that follows). A modern person, in order to be (1)**successful**, must keep studying (2)**throughout** (3)**their** entire life, (4)**mastering** new fields of knowledge. A – pronoun B – adjective C – verb form (participle) D – verb form (adverbial participle) E – preposition

Due to these challenges, the research community has shifted its focus toward developing smaller language models that maintain competitive performance levels [13]. This ongoing effort includes advancements in model training, such as promising parameter-efficient fine-tuning methods (PEFT) that significantly reduce the number of trainable parameters [14], alongside innovative prompting techniques that augment input with instructions to boost performance [15].

By employing these strategies, smaller models can be fine-tuned to approach or even match the performance of their larger competitors in English tasks while mitigating the computational demands [16]. Consequently, there is a growing interest in exploring whether these efficient models can be improved in low-resource setups for underrepresented languages to achieve comparable performance on specific tasks, thereby democratizing access to AI capabilities.

This paper explores the efficacy of fine-tuning compact open-source language models, specifically LLaMA and Gemma, combining PEFT and prompt tuning methods to enhance performance on Ukrainian exam tasks.

II. RELATED WORKS

A. COMPACT LANGUAGE MODELS

Compact language models have gained attention due to their ability to deliver robust performance while requiring fewer computational resources than larger models. With advancements in mobile computing, these compact yet powerful models are increasingly favored for edge device deployment [17]. They offer enhanced privacy and reduced network dependency, making them an attractive option for a wide range of applications. Notable among these compact LLMs are the Gemma 2 and LLaMA 3 model families.

Google's Gemma 2 open-source models are decoder-only large language models designed for text-to-text generation tasks. They are available in multiple parameter sizes, specifically 2 billion (2B), 9 billion (9B), and 27 billion (27B) parameters. The architecture introduces several technical modifications to the Transformer framework, such as interleaving local-global and group-query attention, contributing to improved performance and efficiency [18].

Gemma 2 models have demonstrated exceptional benchmark results across various natural language processing tasks. Notably, these models outperform some larger open models, showcasing their efficiency and effectiveness despite a relatively small parameter count. The instruction-tuned variants of Gemma 2 are reliable at following user prompts and generating coherent, contextually relevant responses [18].

The LLaMA series of open-source LLMs, developed by Meta, has seen significant advancements with the introduction of LLaMA 3, LLaMA 3.1, LLaMA 3.2, and LLaMA 3.3 models. These iterations have progressively enhanced capabilities, model sizes, and functionalities to support diverse AI applications.

Released in April 2024, LLaMA 3 marked a substantial upgrade in Meta's language model offerings. It was introduced in two parameter sizes: 8 billion (8B) and 70 billion (70B). The 70B model was trained on approximately 15 trillion tokens, enabling it to outperform competitors like Gemini Pro 1.5 and Claude 3 Sonnet on various benchmarks [19].

In July 2024, Meta released LLaMA 3.1, expanding the model sizes to include 8B, 70B, and a new 405 billion (405B) parameter model. The 405B model featured an extended context window of up to 128,000 tokens, allowing it to process longer inputs effectively. LLaMA 3.1 aimed to boost efficiency, addressing the limitations of its predecessor [20].

The introduction of LLaMA 3.2 in September 2024 brought significant advancements, particularly in multimodal processing. This version included models with 1B, 3B, 11B, and 90B parameters suitable for various use cases. The 11B and 90B parameter models were designed for joint text and image tasks, while the 1B and 3B models were optimized for deployment on edge devices, supporting real-time processing [21].

B. EFFICIENT FINE-TUNING OF LLM

Full-parameter fine-tuning of large language models can demand substantial computational resources, especially for tasks with long input or output sequences. Multiple parameter-efficient fine-tuning techniques, such as Adapter-based tuning, Prefix tuning, and Low-Rank Adaptation (LoRA), have been developed to address this. With adapter-based tuning, small, trainable layers are being added between the frozen layers of a pre-trained model. This dramatically reduces the number of resources required for the model fine-tuning, while mitigating the issues of forgetting knowledge acquired during pre-training [22]. Prefix tuning also freezes the model and learns a small set of task-specific continuous vectors (prefix), approaching full fine-tuning performance with a minimal number of added prefix parameters [23].

A more recent and elastic technique that consistently achieves full fine-tuning performance with minimal effort in hyperparameter tuning is Low-Rank Adaptation [24]. LoRA reduces the number of trainable parameters by introducing trainable low-rank matrices into each layer of the Transformer architecture, allowing for efficient adaptation of pre-trained models to specific tasks without full model retraining [25]. In addition to high quality and simple parameter selection, this approach also provides efficient inference without additional latency, as the tuned adapter weights can be merged back into the original model after training.

The quantization method is another option to enhance the efficiency of model training or inference. Quantization reduces the precision of the model's weights (e.g., from 32-bit to 8-bit or 4-bit), thereby decreasing memory usage and increasing computation speed. This process can be applied post-training (Post-Training Quantization, or PTQ), which is simple but reduces inference latency at a cost of accuracy degradation [26], or during training (Quantization-Aware Training, or QAT), which is more complex but usually preserves performance [27].

The combination of LoRA and quantization methods has led to significant advancements in model performance. For example, QLoRA enables fine-tuning a 65-billion parameter model on a single 48GB GPU by quantizing the base model to 4-bit and then using LoRA to fine-tune on top of the quantized weights [28]. This technique preserves full 16-bit fine-tuning task performance while being way more memory-efficient. These advancements make deploying sophisticated LLMs in environments with limited computational resources feasible.

C. PROMPTING TECHNIQUES

Prompting techniques have become crucial tools for effectively guiding large language models to perform a wide range of natural language processing tasks and produce the desired output. These methods enable users to configure LLMs for specific behavior without modifying their internal parameters, making them suitable for various applications in low-resource environments. A list of common strategies includes the following.

1. Zero-shot prompting, where the model is given a task description without any examples and is expected to generate the correct output based solely on the prompt [15], leverages the model's pre-existing knowledge to handle tasks on which it has not explicitly been trained. Although it is a simple technique, its performance can be unreliable, particularly on complex reasoning tasks or in languages underrepresented in the pre-training data.

2. Few-shot prompting involves providing the model with a few input-output examples within the prompt to illustrate the task, enabling it to infer and apply the desired pattern to new inputs and typically improving robustness and accuracy [29]. The main limitations of Few-Shot Prompting include the extensive use of the model's limited context window and its high sensitivity to the quality and format of selected examples.

3. Chain-of-thought (CoT) prompting encourages the model to decompose complex problems into a series of intermediate reasoning steps before printing the final answer. This method enhances the model's ability to perform tasks that require logical reasoning and multi-step problem-solving [30]. Despite being an effective strategy, CoT significantly increases the length of the generated output, thus resulting in higher computational costs and increased inference latency. Key variations of this technique include zero-shot CoT, which adds a simple phrase like "think step-by-step" to the prompt, and few-shot CoT, where the provided examples contain detailed reasoning steps.

4. Instruction prompting empowers the model with explicit instructions or guidelines on how to approach a task. Clear and detailed instructions can significantly improve the model's performance by aligning its outputs with user expectations. This technique is the foundation for instruction fine-tuning, a common training phase for modern language models, where they learn to follow diverse user commands from a large dataset of instruction-answer pairs.

5. Generated knowledge prompting forces the model to generate relevant background information before addressing the main task. By first generating this context, the model is generally able to provide more robust and contextually appropriate responses [31]. However, the primary risk is that the model may hallucinate, generating plausible-sounding but incorrect knowledge, which leads to an incorrect final answer.

6. Self-consistency technique is an extension of the chain-of-thought that generates multiple, diverse reasoning paths for the same prompt and then selects the most frequent or "consistent" final answer. This majority-voting approach makes the model's reasoning more robust and less sensitive to arithmetic errors [32]. Its primary weakness is the significant increase in computational cost and time to answer, as it requires running the same prompt multiple times to get a single answer.

7. Tree-of-thoughts is another advanced extension of CoT. With this technique, instead of exploring a single reasoning chain, the model explores multiple different reasoning paths simultaneously [33]. This empowers tree-of-thoughts to solve complex planning or search problems that standard chain-of-thought cannot. However, this power comes with a very high computational cost and implementation complexity.

In general, prompt engineering is a crucial skill for building LLM-powered solutions. It can effectively guide LLMs toward improved generalization and reduced hallucinations, particularly for underrepresented languages and complex problem domains. Furthermore, the strategies mentioned are not mutually exclusive. This research demonstrates that techniques like chain-of-thought and generated knowledge prompting can be combined with parameter-efficient fine-tuning to create highly specialized and efficient models.

D. EXPLAINABLE AI

Explainable Artificial Intelligence is a set of information technologies, models, and methods that help users understand and trust the results produced by machine learning algorithms. Some simple models, such as regression or decision trees, can be explained without additional effort. In earlier knowledge-processing approaches, explanations were provided based on the fragments of knowledge used to obtain prediction results [34]. Modern machine learning methods, such as deep neural networks, are often viewed as "black boxes" due to sophisticated inner workings that are hard to interpret. However, even for these complex models, there are now model-agnostic methods and frameworks for explainability. These typically involve three stages of explanation: pre-modeling (which includes dataset explorations of all kinds), during modeling (where explanations become part of the model's internal functioning), and post-modeling (providing explanations for the prediction results) [35].

Attention visualization is another valuable tool for Transformer-based architectures, especially in natural language processing tasks, where it can highlight the input segments with a high effect on the model’s outputs. However, an even more promising strategy is the chain-of-thought prompting. CoT not only improves the accuracy of predictions but also explicitly presents intermediate steps to unveil the intuition behind any intermediate decision. This detailed explanation simplifies a deeper evaluation of the model’s performance, allowing users to verify that the reasoning aligns with domain-specific rules and principles. This interpretability is crucial for various applications – from academic assessments to healthcare diagnostics – where understanding the motivation behind a decision is no less important than the decision itself.

E. SOLVING EXAM TASKS WITH LLM

The application of LLMs to standardized exam tasks serves as a vital benchmark for their reasoning abilities. For the English language, benchmarks like MMLU, GSM8K, and BIG-Bench provide comprehensive datasets for evaluating model performance on academic examinations:

- MMLU (Massive Multitask Language Understanding) benchmark assesses a model's knowledge and reasoning abilities across over 57 tasks spanning diverse academic disciplines, including mathematics, history, and literature [3];
- GSM8K (Grade School Math 8K) is a widely used benchmark for evaluating multi-step reasoning and arithmetic capabilities, consisting of 8,000 math problems designed to test logical deduction and numerical accuracy [36];
- BIG-Bench (Beyond the Imitation Game Benchmark) – a large-scale benchmark featuring over 200 diverse tasks, such as logic, mathematics, common sense reasoning, and language generation, aimed at pushing models to exhibit deeper cognitive understanding and reasoning [37].

Beyond these foundational benchmarks, other research has explored language model performance on high-school exams in different contexts. For instance, the cross-lingual EXAMS dataset established a strong baseline for scientific question answering with early foundational encoder-only models assessed in 16 languages and 24 subjects from high school examinations [38]. With the advancements of language models, further research has centered on their zero-shot capabilities, demonstrating superficial performance of LLMs in high-resource English language comprehension compared to an average student [39]. For low-resource setups, evaluation of generative models in Latvian centralized exams for school graduates has highlighted the minor difference between leading open-weight and proprietary LLMs [40].

Another notable area of research in the educational domain is the automatic review of human-written answers to assessments. At first, this task may seem unrelated to exam problem-solving, due to its focus on generating rationales justifying the grades assigned to students' responses. However, the model must be aware of the correct solution to provide fair feedback on student answers. Recent studies indicate that combining few-shot or chain-of-thought prompting strategies with contextual item stems and rubrics significantly improves the quality of assessments [41].

As for the Ukrainian language, ZNO-Eval benchmark with real exam tasks from Ukraine's standardized educational testing system, including the External Independent Evaluation and the National Multi-subject Test, comprises single-answer options, matching, correct sequence, and open-ended questions across diverse subjects, delivering a thorough analysis of proprietary LLMs' reasoning capabilities in Ukrainian [12].

At the same time, the UNLP 2024 Shared Task initiative made significant contributions to the benchmarking of open-weight models [42]. This initiative aimed to support the development of models with a deep understanding of the Ukrainian language, literature, and history. It showcased fine-tuning results for numerous promising models and strategies, highlighting advancements in adapting LLMs for Ukrainian-specific tasks [43].

The ZNO-Vision benchmark further extends the evaluation of large language models to multimodal contexts by incorporating over 4,300 expert-crafted questions spanning 12 academic disciplines, including mathematics, physics, chemistry, and humanities [44]. This dataset includes visual elements, enabling the assessment of models' capabilities in handling both text and images.

However, both the UNLP Shared Task and ZNO-Vision evaluations, much like the broader EXAMS dataset, focused solely on questions with a single correct answer. While this prior work provides crucial context, it leaves a gap in understanding how models handle higher-complexity problems requiring structured output in low-resource languages. In contrast, ZNO-Eval tasks involving matching or correct sequences offer a deeper test of reasoning skills. Therefore, these tasks provide a valuable opportunity to investigate whether parameter-efficient fine-tuning can unlock the specialized reasoning required for complex exam formats.

F. THE PURPOSE OF THE RESEARCH

The primary aim of this work is to increase LLM performance on complex Ukrainian language exam tasks in a low-resource setup by employing parameter-efficient chain-of-thought fine-tuning. An important aspect of this research is to check whether, under resource constraints, enhanced fine-tuning and prompting methods can yield performance levels that rival those of larger proprietary models, ultimately advancing the application of cutting-edge information technologies in software engineering for the educational domain.

This research includes the following tasks:

- development of a comprehensive baseline with parameter-efficient fine-tuning of selected open-source language models on a complete set of Ukrainian language exam problems, including multiple-choice and matching tasks;

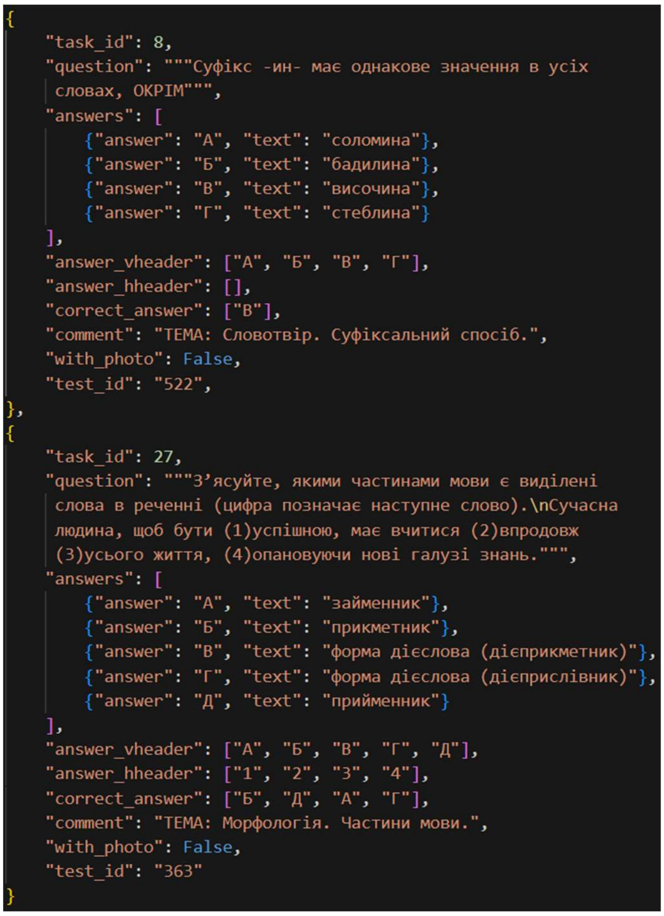
- assessment of the impact of step-by-step reasoning by comparing models tuned solely for single-letter output with those tuned for chain-of-thought generation;

- comparison of the tuned models against leading proprietary and open-weight models.

III. MATERIAL AND METHODS

A. DATA PREPARATION

For training and evaluation, the complete Ukrainian language and literature dataset from the ZNO-Eval benchmark was used. This set consists of single-correct-answer questions and matching tasks, pairing numbered options with lettered options based on the question. The dataset combined 49 ZNO/EIE (External independent evaluation) and NMT exams, totaling 2,746 questions. 32 EIE tests were sampled for training, 13 EIE exams were chosen for validation, and 4 NMT exams were reserved for testing. The NMT exams were chosen for testing to align with the test set used in ZNO-Eval benchmarking and to avoid tasks requiring manual assessment. The training and validation sets included tasks from both the Ukrainian language and literature categories to evaluate generalization capabilities and prevent catastrophic forgetting caused by suboptimal hyperparameter tuning. The test set, however, contained only language tasks. The original ZNO-Eval task schema with the question, answer options, a correct answer, and a comment specifying the task topic was left unchanged (Fig. 1).



```
{
    "task_id": 8,
    "question": """Суфікс -ин- має однакове значення в усіх
    словах, ОКРІМ""",
    "answers": [
        {"answer": "А", "text": "соломина"},
        {"answer": "Б", "text": "бадилина"},
        {"answer": "В", "text": "височина"},
        {"answer": "Г", "text": "стеблина"}
    ],
    "answer_vheader": ["А", "Б", "В", "Г"],
    "answer_hheader": [],
    "correct_answer": ["В"],
    "comment": "ТЕМА: Словотвір. Суфіксальний спосіб.",
    "with_photo": False,
    "test_id": "522",
},
{
    "task_id": 27,
    "question": """З’ясуйте, якими частинами мови є виділені
    слова в реченні (цифра позначає наступне слово).\nСучасна
    людина, щоб бути (1)успішною, має вчитися (2)впродовж
    (3)усього життя, (4)опановуючи нові галузі знань.""",
    "answers": [
        {"answer": "А", "text": "займенник"},
        {"answer": "Б", "text": "прикметник"},
        {"answer": "В", "text": "форма дієслова (дієприкметник)"},
        {"answer": "Г", "text": "форма дієслова (дієприслівник)"},
        {"answer": "Д", "text": "прийменник"}
    ],
    "answer_vheader": ["А", "Б", "В", "Г", "Д"],
    "answer_hheader": ["1", "2", "3", "4"],
    "correct_answer": ["Б", "Д", "А", "Г"],
    "comment": "ТЕМА: Морфологія. Частини мови.",
    "with_photo": False,
    "test_id": "363"
}
```

Figure 1. ZNO-Eval schema for sample tasks from Table 1

Prior to sampling, the dataset was cleaned by removing duplicate tasks (381), paraphrased tasks (52), tasks without answers (4), tasks missing a topic (48), and tasks containing photos in question or answer options (97). This preprocessing resulted in a final dataset of 1,740 tasks for training, 292 tasks for validation, and 108 tasks for testing.

B. TOPIC-GUIDED CHAIN-OF-THOUGHT FINE-TUNING

ZNO-Eval benchmark and baseline evaluations conducted in this research demonstrate that language models often struggle with complex, reasoning-intensive tasks that require structured outputs, especially when dealing with the intricate logic of Ukrainian language tasks [12]. Utilizing the chain-of-thought for prompting and fine-tuning may enhance reasoning abilities [30]. This approach also reveals intermediate thinking steps, which can be audited and analyzed. However, chain-of-thought is time-consuming and computationally demanding during both inference and full-parameter fine-tuning stages. This technique substantially increases the length of the generated output, leading to higher memory consumption and increased generation latency.

Lower-precision parameter-efficient fine-tuning techniques like QLoRA, which leverage 4-bit quantization, have become an effective solution to the resource problem. However, the reduction in resource consumption can lead to reasoning instability, particularly in smaller models [45]. For instance, hallucinations at the beginning of a reasoning chain can cause an "accumulated error" effect, where a single incorrect step fails the entire reasoning process, leading to a completely different final answer.

To address the problem of achieving robust reasoning in a low-resource setting, this research proposes an enhanced parameter-efficient fine-tuning method. This method distills expert-level reasoning capabilities into compact, 4-bit quantized language models by fine-tuning on a structured, multi-part target. Rather than simply outputting an answer word or letter often seen in zero-shot prediction setups, the model is trained to generate, in sequence, two components detailed below.

1. Task topic (e.g., "TOPIC: Morphology. Parts of speech"). This component acts as a form of generated knowledge, but it is based on expert-provided ground truth, not model-hallucinated facts. This step explicitly guides the model to recall and apply the correct domain-specific rules before attempting to solve the problem, narrowing the solution space and reducing the chance of early-stage hallucinations that could lead to wrong answers.

2. Step-by-step solution. The model is trained to generate not only the task topic but also the complete reasoning path, including the final answer. Chain-of-thought here provides the intermediate steps needed for pairwise alignment and format-compliant outputs.

By forcing the model to answer both the "what" (the topic) and the "why" (the reasoning), this method aims to build robust and interpretable reasoning capabilities within a low-resource, 4-bit quantized fine-tuning setup.

For each exam task in the training dataset, expert-curated topics and step-by-step solutions were extracted from the Osvita.ua portal [46], which provides educational materials and exam resources along with commentary written by subject-matter specialists. Table 2 illustrates a sample task topic with its detailed step-by-step solution.

Table 2. Topic and solution from the Osvita.ua portal for sample tasks from Table 1 with English translation

Step-by-step solution	English translation of step-by-step solution
Коментар ТЕМА: Словотвір. Суфіксальний спосіб. Завдання перевіряє ваше вміння розпізнавати вивчені способи словотвору та аналізувати лексичне значення слова. В українській мові за допомогою суфікса -ин- утворюють значну кількість іменників жіночого роду I відміни. Це слова на позначення частин рослини (бадилина, стеблина, соломина), а також на позначення території, рельєфу (височина). **Відповідь – В.**	Comment TOPIC: Word formation. Suffix-Based method. This task tests your ability to recognize common word-formation processes and to analyze a word's lexical meaning. In Ukrainian, the suffix -ин- is used to create many first-declension feminine nouns. These words either refer to plant parts (бадилина, стеблина, соломина) or to geographical features/terrain (височина). **Answer – C.**
Коментар ТЕМА: Морфологія. Частини мови. Завдання перевіряє ваше вміння правильно визначати частини мови. Необхідно бути дуже уважним, тому що частиномовна приналежність конкретного слова часто залежить від контексту. До слова успішною можна поставити питання якою?, воно вказує на ознаку. Це **прикметник**. До слова впродовж не можна поставити питання, воно лише служить для зв'язку слова життя з іншими в реченні. Це **прийменник**. До слова усього можна поставити питання якого?, але воно лише вказує на ознаку, не називаючи її. Це **займенник**. А слово опановуючи відповідає на питання що роблячи?, указує на додаткову дію. Це особлива форма дієслова **дієприслівник**. **Відповідь – БДАГ.**	Comment TOPIC: Morphology. Parts of speech. This task tests your ability to correctly identify parts of speech. It's important to be very attentive, because a word's part of speech often depends on the context. You can ask "якою?" ("which one?") about "успішною" ("successful"), indicating a quality. That makes it an **adjective**. You cannot form a question for "впродовж" ("throughout"); it simply links the word "життя" ("life") to other parts of the sentence. Therefore, it is a **preposition**. You can ask "якого?" ("which one?") about "усього" ("all of"), but it only points to a characteristic without naming it. Hence, it is a **pronoun**. The word "опановуючи" ("mastering" / "while mastering") answers "що роблячи?" ("while doing what?"), indicating an additional action. It is a special verb form called an **adverbial participle**. **Answer – BEAD.**

As shown in the table above, after CoT tuning, the model is expected to generate a relevant hierarchical topic, prefixed with the keyword "ТЕМА:" ("TOPIC:"), followed by a detailed step-by-step solution. The solution includes a review of all answer options or pairs for the exam task, concluding with the keyword "Відповідь:" ("Answer:") and providing either a single answer letter for multiple-choice questions or a sequence of number-letter pairs for matching tasks. This structured approach ensures that the fine-tuned model delivers interpretable and accurate responses while maintaining alignment with task-specific requirements.

C. DATA CONTAMINATION AND LEAKAGE

Data contamination and leakage occur when information from the evaluation dataset inadvertently influences model training, leading to polluted performance metrics [47]. This problem questions the reliability of model evaluation, as it does not accurately reflect its ability to generalize to unseen data. Contamination, common for large language models trained on billions of texts, can arise from various sources, such as shared content between datasets or pre-training on datasets containing evaluation tasks.

In this research, two types of data contamination and leakage were addressed. Pre-training data contamination explores the possibility that the large language model was pre-trained on test-exam tasks. However, this issue is mitigated by several factors. The availability of webpages with Ukrainian exam data is limited, and Ukrainian was not a primary language in the LLM's pre-training dataset. Furthermore, in most cases, the correct answer or problem solution is not directly available alongside the question definition. Accessing the solution often requires additional actions, such as logging in or revealing answers embedded as images rather than text.

To further reduce the impact of potential contamination on evaluation results, the answer numbers, letters, and texts were shuffled for the test set. This measure prevents straightforward answer memorization from contaminating the results.

The second type aims to check whether some tasks within the dataset contained exact or partial matches of questions or answer options across training, validation, and test sets. These duplicates were identified based on matching questions or answer option text in case they were not common generic statements. Generic statements, such as "match options on the left with texts on the right" for questions or part-of-speech keywords for answers, were excluded from duplicate identification.

All duplicates between the validation and training sets were removed from the validation set. Similarly, duplicates between the test set and either the training or validation sets were removed from the train/validation to ensure that the test set remained unchanged. Final manual sample screening identified 40 tasks in the validation set and 12 tasks in the test set that contained rephrased questions or answer options. Although these instances are not exact duplicates, they were removed to prevent data leakage and minimize any potential impact on the evaluation scores.

D. MODEL SELECTION

To align with the low-resource goal of this research, the selection of models was limited to compact options that could be efficiently trained on a single A100 GPU with 80 GB of VRAM. The chosen models include Meta's LLaMA 3.1 with 8 billion parameters, LLaMA 3.2 with 3 billion parameters, and Gemma 2 with 9 billion parameters. These models were selected due to their balance between performance and computational efficiency, making them suitable for resource-constrained environments.

Each model comes in two versions: a base pre-trained and a fine-tuned with instructions to follow user commands in a chat-like manner. This research focuses solely on instruction-tuned versions since pre-trained models did not provide any improvement during initial evaluations.

To further optimize training and inference processes, all models were quantized to 4-bit with the Bits and Bytes library [48]. This quantization significantly reduced memory usage, enabling faster training and inference. The combination of compact architecture and quantization ensured efficient use of computational resources, allowing for effective experimentation under low-resource conditions.

E. MODEL TUNING AND EVALUATION

Parameter-efficient fine-tuning was performed on the selected instruction-tuned models ("it" in the model name) versions, using two variations: one with the correct answer represented as a letter or a sequence of letters, and the other incorporating the proposed chain-of-thought approach with and without topics. Models were fine-tuned over four epochs, with a learning rate of 3e-04, a gradient accumulation of 4 to mimic large batch sizes on GPUs, and checkpoints saved after each epoch. The best checkpoint was identified based on the validation metric that produced the highest overall score on validation exams.

The loss was not used as the validation metric because it is based on the model's perplexity, which does not account for the importance of generating the correct answer letter. Perplexity treats all characters in the generation equally and does not consider the variability in phrasing step-by-step solutions. Instead, validation accuracy, calculated as the sum of all scores on the validation exams, was used to find the best checkpoint.

Data preparation, model configs, and PEFT scripts are available at github.com/NLPForUA/ZNO. Table 3 shows all promising experimental parameters.

Table 3. Experiment parameters

Model	Tuning	Parameters, billions	Trained parameters, millions	Batch size	Accumulation
Tuned for answer letter generation					
LLaMA-3.2-3B-it-tune-al	letter	3	22	8	4
LLaMA-3.1-8B-it-tune-al	letter	8	44	4	4
Gemma-2-9B-it-tune-al	letter	9	52	4	4
Tuned for chain-of-thought (step-by-step solution) generation					
LLaMA-3.2-3B-it-tune-cot	solution	3	22	8	4
LLaMA-3.1-8B-it-tune-cot	solution	8	44	4	4
Gemma-2-9B-it-tune-cot	solution	9	52	4	4
Tuned for chain-of-thought (topic and step-by-step solution) generation					
LLaMA-3.2-3B-it-tune-cot-wt	topic + solution	3	22	8	4
LLaMA-3.1-8B-it-tune-cot-wt	topic + solution	8	44	4	4
Gemma-2-9B-it-tune-cot-wt	topic + solution	9	52	4	4

For evaluation, baseline scores were established using random guessing and zero-shot evaluations of models without CoT output. The evaluation used greedy decoding with a maximum generation length of 2,048 tokens. Generated answers were extracted from the last occurrence of the "Відповідь:" ("Answer:") keyword. The scoring approach followed the same rules for both EIE and NMT exams. Multiple-choice questions were scored 1 point for each correct prediction, while matching tasks were scored up to 4 points, with 1 point awarded for each correct logical pair. For single-answer questions, a score of zero was given if multiple letters were generated, even if the first answer was correct. The score for the matching task was also zeroed if the response contained more than four answer letters, motivating confident solution generation. This methodology ensured consistent evaluation across all models and tasks.

IV. RESULTS

In general, the experimental results prove the effectiveness of parameter-efficient fine-tuning combined with quantization for compact open-source models. For all configurations, tuned models demonstrated substantial improvements over the baseline, with joint topic generation and step-by-step reasoning contributing moderately to performance gains.

The added benefit of chain-of-thought tuning (LLaMA and Gemma models with "cot" suffix) becomes clearer when applied to more complex tasks, including matching and literature assignments (scores shown in parentheses for literature tasks in Table 4). In these scenarios, the implementation of step-by-step reasoning enhances the steerability and clarity of the model’s thought process, making it easier to follow the logic it employs to arrive at conclusions. However, despite these gains, the validation set did not consistently show anticipated improvement when comparing chain-of-thought to letter-only generation. Several factors appear to affect the result. Firstly, the validation set primarily consists of older exam tasks with no answer option shuffling, thus increasing the chance of data contamination. Secondly, the approach taken to remove duplicate and rephrased tasks has inadvertently led to an uneven distribution of task types and topics. Some appear only once or twice, whereas others are overrepresented. Lastly, adapters were merged with base models for validation scoring due to time and cost considerations. This could lead to a score drop for CoT models.

Nevertheless, the validation scores remain valuable for selecting the optimal training epoch. It has been empirically observed that higher single-answer, matching, and total validation scores directly correlate with better performance on a more representative test set. In contrast to the validation set, the test data includes more recent exams with answer options shuffling and a fair balance of question types and topics.

Detailed experiment results are demonstrated in Tables 4 and 5. All models are available at *huggingface.co/NLPForUA*.

Table 4. Evaluation results on the validation set with 4-bit quantization

Model Name	Generates	Scores for language tests			Scores for language and literature		
		Single answer	Matching	Total	Single answer	Matching	Total
Max possible score	-	233	72	305	260 (+27)	88 (+16)	348
Random guess	letter	53.3	14.4	67.7	58.8 (+5.5)	17.6 (+3.2)	76.4
Baseline: zero-shot answer letter generation							
LLaMA-3.2-3B-it	letter	0	1	1	1 (+1)	1 (+0)	2
Qwen2.5-7B-it	letter	52	5	57	57 (+5)	8 (+3)	65
LLaMA-3.1-8B-it	letter	66	10	76	71 (+5)	11 (+1)	82
Gemma-2-9B-it	letter	31	16	47	36 (+5)	18 (+2)	54
Qwen2.5-14B-it	letter	69	16	85	81 (+12)	19 (+3)	100
Gemma-2-27B-it	letter	**79**	**20**	**99**	**88** (+9)	**22** (+2)	**110**
Qwen2.5-32B-it	letter	40	12	52	48 (+8)	16 (+4)	64
LLaMA-3.3-70B-it	letter	56	15	71	64 (+8)	18 (+3)	82
Qwen2.5-72B-it	letter	61	12	73	74 (+13)	14 (+2)	88
Reasoning models baseline: zero-shot chain-of-thought generation							
DeepSeek-R1 LLaMA-8B	solution	9	0	9	11 (+2)	0 (+0)	11
DeepSeek-R1 Qwen-14B	solution	25	13	38	35 (+10)	13 (+0)	48
DeepSeek-R1 Qwen-32B	solution	43	**29**	72	**51** (+8)	**29** (+0)	**80**
LLaMA 3.2 3B							
LLaMA-3.2-3B-it-tune-al	letter	57	16	73	**65** (+8)	17 (+1)	**82**
LLaMA-3.2-3B-it-tune-cot	solution	54	17	71	63 (+9)	**18** (+1)	81
LLaMA-3.2-3B-it-tune-cot-wt	topic+solution	53	8	61	60 (+7)	13 (+5)	73
LLaMA 3.1 8B							
LLaMA-3.1-3B-it-tune-al	letter	74	27	101	82 (+8)	31 (+4)	113
LLaMA-3.1-8B-it-tune-cot	solution	82	28	110	**94** (+12)	32 (+4)	126
LLaMA-3.1-8B-it-tune-cot-wt	topic+solution	81	35	116	91 (+10)	**38** (+3)	**129**
Gemma 2 9B							
Gemma-2-9B-it-tune-al	letter	104	37	141	**118** (+14)	41 (+4)	**159**
Gemma-2-9B-it-tune-cot	solution	96	41	137	109 (+13)	**44** (+3)	153
Gemma-2-9B-it-tune-cot-wt	topic+solution	94	37	131	110 (+16)	39 (+2)	149

Table 5. Evaluation results on the test set with 4-bit quantization

Model Name	Generates	Total scores			Total scores after merge		
		Single answer	Matching	Total	Single answer	Matching	Total
Max possible score	-	92	64	156	-	-	-
Random guess	letter	20.25	12.78	33.03	-	-	-
Baseline: zero-shot answer letter generation							
LLaMA-3.2-3B-it	letter	0	4	4	-	-	-
Qwen2.5-7B-it	letter	26	5	31	-	-	-
LLaMA-3.1-8B-it	letter	25	7	32	-	-	-
Gemma-2-9B-it	letter	21	21	42	-	-	-
Qwen2.5-14B-it	letter	25	16	41	-	-	-
Gemma-2-27B-it	letter	**30**	24	**54**	-	-	-
Qwen2.5-32B-it	letter	18	**26**	44	-	-	-
LLaMA-3.3-70B-it	letter	25	13	38	-	-	-
Qwen2.5-72B-it	letter	18	15	33	-	-	-
Reasoning models baseline: zero-shot chain-of-thought generation							
DeepSeek-R1 LLaMA-8B	solution	4	1	5	-	-	-
DeepSeek-R1 Qwen-14B	solution	16	21	37	-	-	-
DeepSeek-R1 Qwen-32B	solution	**22**	**25**	**47**	-	-	-
LLaMA 3.2 3B							
LLaMA-3.2-3B-it-tune-al	letter	24	11	35	**27**	10	37
LLaMA-3.2-3B-it-tune-cot	solution	18	10	28	16	14	30
LLaMA-3.2-3B-it-tune-cot-wt	topic+solution	24	**15**	**39**	14	5	19
LLaMA 3.1 8B							
LLaMA-3.1-3B-it-tune-al	letter	25	12	37	**30**	**17**	**47**
LLaMA-3.1-8B-it-tune-cot	solution	19	13	32	26	13	39
LLaMA-3.1-8B-it-tune-cot-wt	topic+solution	26	15	41	28	14	42
Gemma 2 9B							
Gemma-2-9B-it-tune-al	letter	**33**	23	56	41	22	63
Gemma-2-9B-it-tune-cot	solution	37	27	**64**	28	**29**	57
Gemma-2-9B-it-tune-cot-wt	topic+solution	29	**30**	59	28	26	54

The random guessing baseline, selecting one random answer out of all provided options for questions with a single correct answer and constructing a sequence of random letters for matching tasks, achieved a total test score of 33.03, with 20.25 points on single-answer questions and 12.78 points on matching tasks. The overall performance of baseline LLaMA 3.1 and 3.2 models reflects the underrepresented nature of the Ukrainian language, as the former fails to provide any meaningful answer (total score of 4) and the latter struggles to surpass random guessing in matching tasks (7 vs 12.78 points). At the same time, Gemma-2-9B demonstrated high robustness without any fine-tuning, securing 42 in total. In comparison, its

3 times larger "relative" became a leader with 54 points.

An important consideration is the effect of merging a 4-bit LoRA adapter with the base model. Directly merging in 4-bit often degrades prediction quality due to rounding errors. Another approach, a full precision merge with a subsequent quantization, helped mitigate the issue. Interestingly, letter-only models show substantial gains after merging as the impact of numerical artifacts increases with the length of generation.

All instruct models tuned to generate answer letters or sequences of letters (models with the "it-tune-al" suffix) demonstrated reasonable improvements over the baseline. For instance, the LLaMA-3.2-3B model tuned with topics and solutions slightly exceeded the random guessing score (39 vs. 33 points), while LLaMA-3.1-8B-it-tune-al yielded 49.9% improvement with a total score of 47 points, and Gemma-2-9B-tune-al demonstrated the second-best result with 63 points.

Fine-tuning models with step-by-step reasoning solutions in a chain-of-thought manner revealed nuanced outcomes. While the CoT approach substantially improved performance across all matching and literature tasks compared to letter-only solutions, only Gemma-2-9B-it-tune-cot achieved a higher total score with 14.3% and 1.6% gains over the letter-only solution before and after model merge.

A proposed method, prefacing step-by-step reasoning with a generated task topic, helped squeeze the solution space, enhancing both interpretability and accuracy. With this approach, the topic is kind of a knowledge prompt, guiding the model to focus on relevant aspects of the task. That pseudo-knowledge injection increases the likelihood of the domain-specific terms and rules, helping the model "recall" information memorized during pre-training or instruction tuning. Joint topic and step-by-step solution generation ("cot-wt" suffix) allowed to surpass the 30-point bar for matching tasks and secure 18 points for literature with Gemma model, while also showing the highest score of 39 points across all LLaMA-3.2 solutions with 5.4% and 30% increase over letter and CoT models.

Although chain-of-thought enhancements over a letter-only generation are not always dramatic, this approach has unquestionable benefits with improved matching scores and interpretability (Table 6).

Table 6. Solutions for sample tasks by Gemma-2-9B-it-tune-cot-wt and their translations

Generated chain-of-thought solution	English translation of generated chain-of-thought solution
<bos><start_of_turn>user Дайте розгорнуту відповідь на завдання, починаючи з ключового слова "Відповідь:" та використовуючи лише наведені нижче варіанти. Завдання: Суфікс -ин- має однакове значення в усіх словах, ОКРІМ Варіанти відповіді: А - соломина Б - бадилина В - височина Г - стеблина<end_of_turn> <start_of_turn>model Відповідь: ТЕМА: Будова слова. Словотвір. Значущі частини слова: корінь, префікс, суфікс, закінчення Для виконання цього завдання треба пригадати поняття значущих частин слова (морфем), зокрема кореня й суфікса. Суфікс -ин- має різне значення. Так у словах соломина, стеблина, бадилина він позначає відірване від чогось, відбиток, уламки, дрібні частинки чогось. А от у слові височина цей суфікс має значення збірності. Відповідь – В.<end_of_turn>	<bos><start_of_turn>user Provide a detailed answer to the task, starting with the keyword “Відповідь:” and using only the options listed below. Task: The suffix -ин- has the same meaning in the following words EXCEPT Answer options: A – соломина (straw piece) B – бадилина (leafy-stalk piece) C – височина (height / highland) D – стеблина (stem piece) <end_of_turn> <start_of_turn>model TOPIC: Word Structure. Word Formation. Meaningful Parts of the Word: Root, Prefix, Suffix, Ending To complete this task, you need to recall the concept of the meaningful parts of a word (morphemes), particularly the root and the suffix. The suffix -ин- has different meanings. In the words “соломина”, “стеблина”, and “бадилина”, it indicates something torn off, a fragment, remainder, or small part of something. However, in the word “височина”, this suffix signifies collectivity. Answer – B. <end_of_turn>
<bos><start_of_turn>user Дайте розгорнуту відповідь на завдання, починаючи з ключового слова "Відповідь:" та використовуючи лише наведені нижче варіанти. Завдання: З’ясуйте, якими частинами мови є виділені слова в реченні (цифра позначає наступне слово). Сучасна людина, щоб бути (1)успішною, має вчитися (2)впродовж (3)усього життя, (4)опановуючи нові галузі знань. Варіанти відповіді: А – займенник Б – прикметник В – форма дієслова (дієприкметник) Г – форма дієслова (дієприслівник) Д – прийменник<end_of_turn> <start_of_turn>model Відповідь: ТЕМА: Морфологія. Частини мови Завдання перевіряє ваше вміння визначати частини мови. Необхідно бути дуже уважним, тому що частиномовна приналежність конкретного слова часто залежить від контексту. Проаналізуймо кожне із запропонованих слів. Слово **успішною** відповідає на питання яка? та вказує на ознаку предмета. Отже, це прикметник. Слово **впродовж** не відповідає на питання, а лише поєднує слова в реченні. Отже, це прийменник. Слово **усього** відповідає на питання що? і вказує на предмет, але не називає його. Отже, це займенник. А от слово **опановуючи** відповідає на питання що роблячи? та вказує на додаткову дію підмета. Отже, це особлива форма дієслова дієприслівник. Відповідь: 1 – Б, 2 – Д, 3 – А, 4 – Г.<end_of_turn>	<bos><start_of_turn>user Provide a detailed answer to the task, starting with the keyword “Відповідь:” and using only the options listed below. Task: Determine which parts of speech the highlighted words are in the following sentence (the number indicates the word that follows). A modern person, in order to be (1)successful, must keep studying (2)throughout (3)their entire life, (4)mastering new fields of knowledge. Answer options: A – pronoun B – adjective C – verb form (participle) D – verb form (adverbial participle) E – preposition<end_of_turn> <start_of_turn>model Answer: TOPIC: Morphology. Parts of speech This task checks your ability to identify parts of speech. You must be very attentive, because the part of speech for a given word often depends on the context. Let’s analyze each of the proposed words. The word **successful** answers the question “which one?” and indicates a characteristic of an object. So, it is an adjective. The word **throughout** does not answer a question; it merely connects words in the sentence. Therefore, it is a preposition. The word **their** answers the question “whose?” and indicates an object without naming it. So, it is a pronoun. The word **mastering** answers the question “doing what?” and indicates an additional action of the subject. Therefore, it is a special verb form known as an adverbial participle. Answer: 1 – B, 2 – E, 3 – A, 4 – D.<end_of_turn>

Gemma's solution presented above demonstrates several strengths but also has some limitations. In the first task, the model seems to apply a deep linguistic analysis with strong reasoning. However, in the second task, the generated answer explains the reasoning behind each matching decision, clarifying the pairing of specific fragments without going deeper into the morphological aspects behind each answer option. Both answers are correct, so this result is still substantial as less than 50% of graduates select a correct option in tasks like the first problem, and only 28%, on average, strike four out of four in problems similar to the second one [49].

In addition to a comparison between open-weight LLMs, it is also crucial to check how close the obtained solutions are to leading proprietary models widely used by the community and enterprise. Fig. 2 presents a combined result chart of tuned models and zero-shot LLMs.

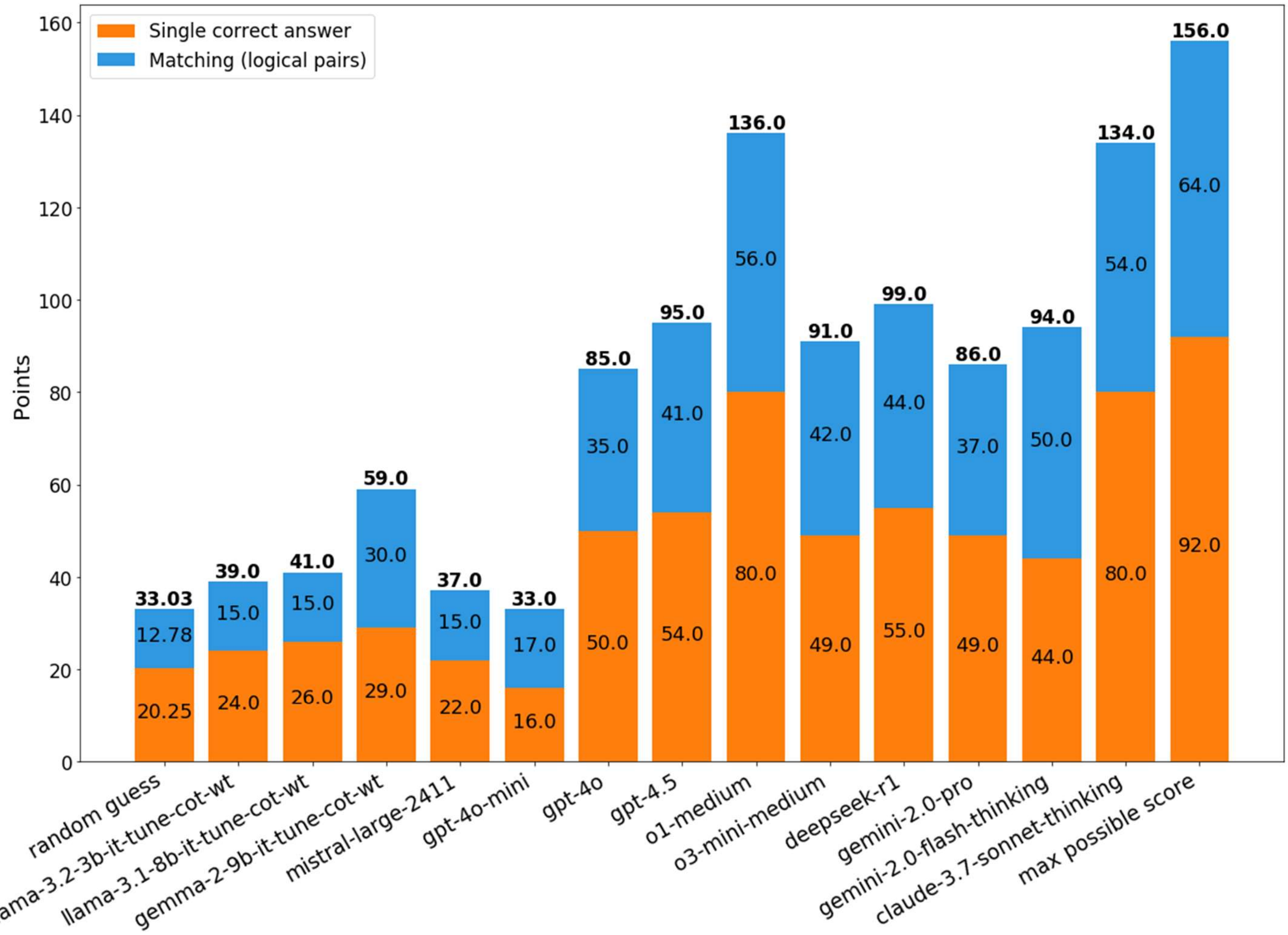


Figure 2. Evaluation results for tuned open-weight and zero-shot proprietary models

Despite these advancements, compact models still cannot reach the performance level of leading reasoning models like OpenAI o1 or Claude 3.7 Sonnet, which benefit from extensive multilingual datasets and demonstrate strong results in complex tasks. However, fine-tuned models highlight that combining parameter-efficient fine-tuning with CoT reasoning could significantly narrow the performance gap in a low-resource setup and even slightly outperform larger LLMs (GPT-4o mini and Mistral Large). Moreover, Gemma secured 30 points for matching tasks, getting relatively close to powerful GPT-4o and Gemini 2.0 Pro models (35 and 37 points).

V. CONCLUSIONS

This research provides several important contributions to the field of natural language processing, particularly for low-resource setups and underrepresented languages. Furthermore, to the best of our knowledge, this work represents the first comprehensive evaluation of large language models on matching tasks for Ukrainian language exams and extends the Ukrainian language exam benchmark with common open-weight and proprietary reasoning models.

The scientific novelty of this research lies in the proposed topic-guided chain-of-thought fine-tuning method, which represents a further development of both parameter-efficient fine-tuning and the chain-of-thought methodology. This method is designed to address reasoning instability in compact, quantized models by training the model to jointly generate two components: a task topic (expert label) that narrows the solution space and aims to reduce early-stage hallucinations, and the complete step-by-step reasoning path. This approach not only improves quality on complex matching tasks compared to standard answer letter generation and chain-of-thought tuning in reasoning-intensive Ukrainian exam tasks for open-weight LLaMA and Gemma models but also underscores the potential for cost-effective alternatives to proprietary LLMs.

The practical significance of the research is the demonstration of how compact models can be optimized to perform well on complex tasks in low-resource environments. By using a single A100 GPU, LoRA, and 4-bit quantization techniques, the work underscores the possibility of training advanced NLP systems in computationally constrained environments. The findings are particularly relevant for underrepresented languages, where access to proprietary models and computational resources may be limited.

The limitation of this research is that the evaluation data size, though representative, is relatively small and may not fully capture the diversity of real-world tasks. Additionally, unavoidable data contamination during pre-training and the limited hyperparameter exploration in the experiments could influence the generalization of the obtained results. Moreover, the use of 4-bit quantization, while beneficial for efficiency, might also introduce subtle degradation in model performance, which requires further exploration.

The prospect for further research is to mitigate the aforementioned limitations by expanding the evaluation dataset to include more diverse tasks, exploring multimodal reasoning capabilities, and experimenting with a broader range of hyperparameters.

VI. ACKNOWLEDGEMENTS

We sincerely appreciate the contributions of Meta and Google for making the LLaMA and Gemma model families openly accessible, empowering advancements in open-source research and language model fine-tuning. Additionally, we extend our gratitude to the Unsloth and vLLM teams for providing an efficient framework for large language model inference in a low-resource setup. Furthermore, special thanks go to Osvita.ua for building a solid platform featuring exam tasks with detailed solutions.

References

[1] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020.

[2] P. Rajpurkar, *The Stanford Question Answering Leaderboard*, 2025, [Online] Available at: https://rajpurkar.github.io/SQuAD-explorer/.

[3] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. X. Song and J. Steinhardt. "Measuring massive multitask language understanding," *Proceedings of the Ninth International Conference on Learning Representations, ICLR 2021*, Virtual Event, Austria, May 3-7, 2021.

[4] B. Romera-Paredes, M. Barekatain, A. Novikov, M. Balog, M. P. Kumar, E. Dupont, F. J. Ruiz, J. S. Ellenberg, P. Wang, O. Fawzi, P. Kohli, A. Fawzi, J. Grochow, A. Lodi, J. Mouret, T. Ringer and T. Yu, "Mathematical discoveries from program search with large language models," Nature, vol. 625, no. 7995, pp. 468 – 475, 2024. https://doi.org/10.1038/s41586-023-06924-6.

[5] OpenAI, *GPT-4o System Card*, 2024, [Online] Available at: https://arxiv.org/abs/2410.21276.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, "Attention Is All You Need," in Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, December 4-9 2017, pp. 5998-6008.

[7] J. Ainslie, J. Lee-Thorp, M. de Jong, Y. Zemlyanskiy, F. Lebrón, S.K. Sanghai, "GQA: Training Generalized Multi-Query Transformer Models from Multi-Head Checkpoints," *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023*, Singapore, December 6-10, 2023, pp. 4895–4901. https://doi.org/10.18653/v1/2023.emnlp-main.298.

[8] J. Kaplan, S. McCandlish, T. Henighan, T.B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, D. Amodei, *Scaling Laws for Neural Language Models*, 2020, [Online]. Available at: https://arxiv.org/abs/2001.08361.

[9] D. Driess, F. Xia, M.S.M. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu, W. Huang, Y. Chebotar, P. Sermanet, D. Duckworth, S. Levine, V. Vanhoucke, K. Hausman, M. Toussaint, K. Greff, A. Zeng, I. Mordatch, P. Florence, "PaLM-E: An Embodied Multimodal Language Model," *Proceedings of the 40th International Conference on Machine Learning, ICML 2023*, Honolulu, Hawaii, USA, July 23-29 2023, pp. 8469-8488.

[10] N.O. Komleva, K.S. Cherneha, B.I. Tymchenko, O.M. Komlevoy, "Intellectual Approach Application for Pulmonary Diagnosis," *IEEE First International Conference Data Stream Mining & Processing (DSMP)*, Lviv, Ukraine, August 23–27, 2016, pp. 48–52. https://doi.org/10.1109/DSMP.2016.7583505.

[11] J. Myung, N. Lee, Y. Zhou, J. Jin, R.A. Putri, D. Antypas, H. Borkakoty, E. Kim, C. Pérez-Almendros, A.A. Ayele, V. Gutiérrez-Basulto, Y. Ibáñez-García, H. Lee, S.H. Muhammad, K. Park, A. Rzayev, N. White, S.M. Yimam, M.T. Pilehvar, N. Ousidhoum, J. Camacho-Collados, A. Oh, "BLEnD: A Benchmark for LLMs on Everyday Knowledge in Diverse Cultures and Languages," *Proceedings of the 38th Conference on Neural Information Processing Systems (NeurIPS 2024)*, Vancouver, Canada, December 10-15 2024.

[12] M.V. Syromiatnikov, V.M. Ruvinskaya, A.S. Troynina, "ZNO-Eval: Benchmarking reasoning capabilities of large language models in Ukrainian," *Informatics. Culture. Technology.*, vol. 1, no. 1, pp. 185–191, 2024. https://doi.org/10.15276/ict.01.2024.27.

[13] DeepSeek-AI, *DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning*, 2025, [Online]. Available at: https://arxiv.org/abs/2501.12948.

[14] Z. Han, C. Gao, J. Liu, J. Zhang, S.Q. Zhang, *Parameter-Efficient Fine-Tuning for Large Models: A Comprehensive Survey*, 2024, [Online]. Available at: https://arxiv.org/abs/2403.14608.

[15] P. Sahoo, A.K. Singh, S. Saha, V. Jain, S.S. Mondal, A. Chadha, *A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications*, 2024, [Online]. Available at: https://arxiv.org/abs/2402.07927.

[16] H.W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S.S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, D. Valter, S. Narang, G. Mishra, A.W. Yu, V. Zhao, Y. Huang, A.M. Dai, H. Yu, S. Petrov, E.H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q.V. Le, J. Wei, "Scaling Instruction-Finetuned Language Models," *Journal of Machine Learning Research*, vol. 25, no. 70, pp. 1-53, 2024.

[17] K. Alizadeh-Vahid, I. Mirzadeh, D. Belenko, K. Khatamifard, M. Cho, C.C. Del Mundo, M. Rastegari, M. Farajtabar, "LLM in a Flash: Efficient Large Language Model Inference with Limited Memory," *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Bangkok, Thailand, August 11–16, 2024, pp. 12562–12584. https://doi.org/10.18653/v1/2024.acl-long.678.

[18] Gemma Team, *Gemma 2: Improving Open Language Models at a Practical Size*, 2024, [Online]. Available at: https://arxiv.org/abs/2408.00118.

[19] LLaMA team, *The Llama 3 Herd of Models*, 2024 [Online]. Available at: https://arxiv.org/abs/2407.21783.

[20] Meta AI, *Introducing Llama 3.1: Our Most Capable Models to Date, 2024*, [Online]. Available at: https://ai.meta.com/blog/meta-llama-3-1/.

[21] Meta AI, *Llama 3.2: Revolutionizing Edge AI and Vision with Open, Customizable Models*, 2024, [Online]. Available at: https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/.

[22] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. de Laroussilhe, A. Gesmundo, M. Attariyan, S. Gelly, *Parameter-Efficient Transfer Learning for NLP*, 2019, [Online]. Available at: https://arxiv.org/abs/1902.00751.

[23] X. Li, P. Liang, *Prefix-Tuning: Optimizing Continuous Prompts for Generation*, 2021, [Online]. Available at: https://arxiv.org/abs/2101.00190.

[24] V. Lialin, V. Deshpande, A. Rumshisky, *Scaling down to scale up: A guide to parameter-efficient fine-tuning*, 2023, [Online]. Available at: https://arxiv.org/abs/2303.15647.

[25] E.J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, *LoRA: Low-Rank Adaptation of Large Language Models*, 2021, [Online]. Available at: https://arxiv.org/abs/2106.09685.

[26] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam, and D. Kalenichenko, "Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018)*, Salt Lake City, USA, June 18-22, 2018, pp. 2704–2713. https://doi.org/10.1109/CVPR.2018.00286.

[27] S. K. Esser, J. L. McKinstry, D. Bablani, R. Appuswamy, and D. S. Modha, "Learned Step Size Quantization," *The Eighth International Conference on Learning Representations (ICLR 2020)*, Online, April 26–May 1, 2020. https://doi.org/10.48550/arXiv.1902.08153.

[28] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, *QLoRA: Efficient Finetuning of Quantized LLMs*, 2023, [Online]. Available at: https://arxiv.org/abs/2305.14314.

[29] M.V. Syromiatnikov, V.M. Ruvinskaya, "UA-LLM: Advancing Context-Based Question Answering in Ukrainian Through Large Language Models," *Radio Electronics, Computer Science, Control*, no. 1, pp. 147-161. 2024. https://doi.org/10.15588/1607-3274-2024-1-14.

[30] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E.H. Chi, Q. Le, D. Zhou, *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models*, 2022, [Online]. Available at: https://arxiv.org/abs/2201.11903.

[31] J. Liu, A. Liu, X. Lu, S. Welleck, P. West, R. Le Bras, Y. Choi, H. Hajishirzi, "Generated Knowledge Prompting for Commonsense Reasoning," *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Dublin, Ireland, May 22–27, 2022, pp. 3154–3169. https://doi.org/10.18653/v1/2022.acl-long.225.

[32] X. Wang, J. Wei, D. Schuurmans, Q. V. Le, E. H. Chi, S. Narang, A. Chowdhery, and D. Zhou, *Self-Consistency Improves Chain of Thought Reasoning in Language Models*, 2023, [Online]. Available at: https://arxiv.org/abs/2203.11171v4.

[33] S. Yao et al., *Tree of Thoughts: Deliberate Problem Solving with Large Language Models*, 2023, [Online]. Available at: https://arxiv.org/abs/2305.10601.

[34] V.M. Ruvinskaya, A.S. Troynina. "Development of information technology for the generation and maintenance of knowledge–oriented control systems," *Eastern-European Journal of Enterprise Technologies*, vol. 2, no. 86, pp. 41–49, 2017. https://doi.org/10.15587/1729-4061.2017.98727.

[35] C. Molnar, *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*, 2025, [Online]. Available at: https://christophm.github.io/interpretable-ml-book/

[36] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, Ł. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, J. Schulman, *Training Verifiers to Solve Math Word Problems*, 2021, [Online]. Available at: https://arxiv.org/abs/2110.14168.

[37] A. Srivastava et al., "Beyond the Imitation Game: Quantifying and Extrapolating the Capabilities of Language Models," *Transactions on Machine Learning Research*, vol. 5, pp. 1–95, 2023. DOI: https://doi.org/10.48550/arXiv.2206.04615.

[38] M. Hardalov, T. Mihaylov, D. Zlatkova, Y. Dinkov, I. Koychev, and P. Nakov, "EXAMS: A Multi-subject High School Examinations Dataset for Cross-lingual and Multilingual Question Answering," *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Online, November 16–20, 2020, pp. 5427–5444. https://doi.org/10.18653/v1/2020.emnlp-main.438.

[39] J. C. de Winter, "Can ChatGPT pass high school exams on English language comprehension?," *International Journal of Artificial Intelligence in Education*, vol. 34, no. 3, pp. 915–930, 2024. https://doi.org/10.1007/s40593-023-00372-z.

[40] R. Darģis, G. Barzdins, I. Skadiņa, N. Gruzitis, B. Saulīte, "Evaluating open-source LLMs in low-resource languages: Insights from Latvian high school exams," *Proceedings of the 4th International Conference on Natural Language Processing for Digital Humanities (NLP4DH 2024)*, Bangkok, Thailand, November 16, 2024, pp. 289–293. https://doi.org/10.18653/v1/2024.nlp4dh-1.28.

[41] G. G. Lee, E. Latif, X. Wu, N. Liu, and X. Zhai, "Applying large language models and chain-of-thought for automatic scoring," *Computers and Education: Artificial Intelligence*, vol. 6, 100213, 2024. https://doi.org/10.1016/j.caeai.2024.100213.

[42] M. Romanyshyn, O. Syvokon, R. Kyslyi, "The UNLP 2024 Shared Task on Fine-Tuning Large Language Models for Ukrainian," *Proceedings of the Third Ukrainian Natural Language Processing Workshop (UNLP) @ LREC-COLING 2024*, Torino, Italia, May 25, 2024, pp. 67–74.

[43] A. Kiulian, A. Polishko, M. Khandoga, O. Chubych, J. Connor, R. Ravishankar, A. Shirawalmath, "From Bytes to Borsch: Fine-Tuning Gemma and Mistral for the Ukrainian Language Representation," *Proceedings of the Third Ukrainian Natural Language Processing Workshop (UNLP) @ LREC-COLING 2024*, Torino, Italia, May 25, 2024, pp. 83–94.

[44] Y. Paniv, A. Kiulian, D. Chaplynskyi, M. Khandoga, A. Polishko, T. Bas, G. Gabrielli, *Benchmarking Multimodal Models for Ukrainian Language Understanding Across Academic and Cultural Domains*, 2024. https://doi.org/10.18653/v1/2025.unlp-1.2.

[45] Z. Li, Y. Su, R. Yang, C. Xie, Z. Wang, Z. Xie, N. Wong, H. Yang, *Quantization Meets Reasoning: Exploring LLM Low-Bit Quantization Degradation for Mathematical Reasoning*, 2025, [Online]. Available at: https://arxiv.org/abs/2501.03035.

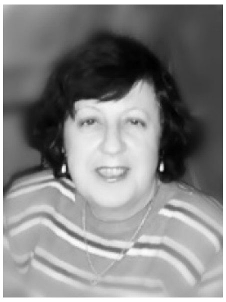
[46] Osvita.ua, *Ukrainian ZNO exams*, 2024, [Online]. Available at: https://zno.osvita.ua/ukrainian/.

[47] S. Balloccu, P. Schmidtová, M. Lango, O. Dušek, *Leak, Cheat, Repeat: Data Contamination and Evaluation Malpractices in Closed-Source LLMs*, 2024. https://doi.org/10.18653/v1/2024.eacl-long.5.

[48] Bitsandbytes Foundation, *bitsandbytes*, 2025, [Online]. Available at: https://github.com/bitsandbytes-foundation/bitsandbytes.

[49] Ukrainian Center for Education Quality Assessment. *Official reports*, 2025, [Online]. Available at: https://testportal.gov.ua/ofzvit/

MYKYTA V. SYROMIATNIKOV, Postgraduate student of the Department of Software Engineering, Odesa Polytechnic National University. Research areas: natural language processing, large language modeling, and deep learning.
ORCID: 0000-0002-0610-3639



VICTORIA M. RUVINSKAYA, PhD, Professor of the Department of Software Engineering, Odesa Polytechnic National University. Research areas: knowledge-based systems, machine learning, algorithms and data structures.
ORCID: 0000-0002-7243-5535

NATALIIA O. KOMLEVA, PhD, Associate Professor, Head of the Department of Software Engineering, Odesa Polytechnic National University. Research areas: data analysis, software engineering, knowledge management.
ORCID: 0000-0001-9627-8530