SALUTE: Benchmarking and Adapting LLMs for the Defense Domain

Hyeongcheol Park[1] **Sumin In**[1] **Suyeon Myeong**[1] **Hogun Park**[2]
Sangmin Kim[3] **Moonhyun Lee**[3] **Daekyeong Park**[3] **Sangpil Kim**[1]*
[1]Korea University, Republic of Korea
[2]Sungkyunkwan University, Republic of Korea
[3]Hanwha Systems, Republic of Korea
[1]{broiron,ism0705,tnduss,spk7}@korea.ac.kr
[2]hogunpark@skku.edu
[3]{smkim0153,moonhyun.lee,daekyeong.park}@hanwha.com

Abstract

Defense is a knowledge-intensive domain that requires precise understanding of specialized terminology, doctrinal concepts, operational procedures, and evolving military events. Although recent work has explored language technologies for military applications, existing efforts remain fragmented: they are often task-specific, rely on limited adaptation pipelines, or lack comprehensive defense-domain evaluation. In this paper, we present **SALUTE**, an end-to-end framework for benchmarking and adapting LLMs for the defense domain. SALUTE integrates Salute-Corpus, a curated corpus from open-access U.S. military doctrine and government documents; Salute-Conv, a grounded instruction dataset from doctrinal sources and decade-long defense news; Salute-Pref, a defense-aware preference dataset; and Salute-Bench, a rigorously filtered benchmark for evaluating defense-domain understanding and reasoning over doctrine and defense news. Based on these resources, we train Salute-LLM through multi-stage post-training with continual pretraining, supervised fine-tuning, and preference alignment. Extensive experiments show that Salute-LLM achieves strong defense-domain performance while retaining competitive general capabilities, demonstrating the effectiveness of SALUTE as an end-to-end framework for defense-domain LLM adaptation.

1 Introduction

Modern defense environments are increasingly shaped by information overload (Meerveld et al., 2023). Defense-relevant information spans doctrine, technical manuals, and rapidly evolving operational news, each requiring precise understanding of specialized terminology, operational context, and domain-specific reasoning. This motivates the development of language models capable of defense-specific understanding and reasoning. However, despite their strong general capabilities, general-purpose LLMs (Grattafiori et al., 2024; Yang et al., 2025) remain limited in specialized domains that require precise terminology, domain-specific background knowledge, and reliable reasoning (Shi et al., 2025). These challenges highlight the need for defense-specialized LLMs and an integrated framework for data construction, model adaptation, and comprehensive evaluation.

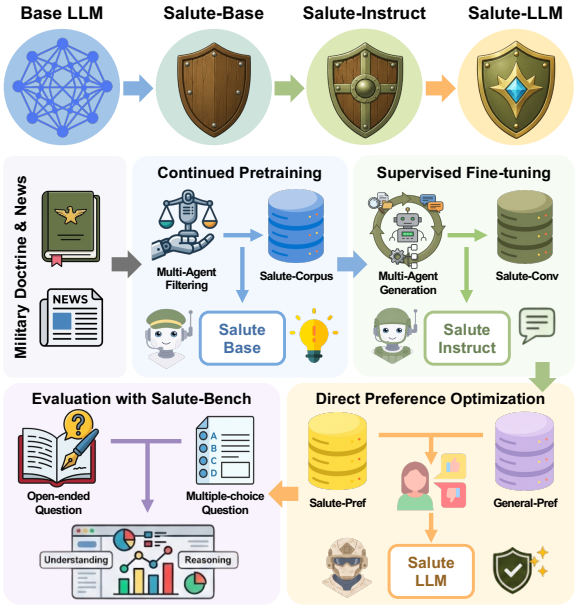

Figure 1: Overview of **SALUTE**, a unified framework for defense-domain LLM adaptation.

Recent work has explored language technologies for defense and military applications, but existing efforts still exhibit three important limitations. **First**, prior resources (Zhu et al., 2024; Palnitkar et al., 2026) often target specific task settings, such as military event extraction or geospatial planning, rather than broad defense-domain language understanding. **Second**, defense-domain LLM efforts (Li et al., 2022; Xue et al., 2024; Ruiz and Sell, 2024; FitzGerald et al., 2025) typically rely on single-stage adaptation and do not provide reproducible pipelines that connect data construction, multi-stage training, and evaluation. **Third**, existing defense-domain benchmarks (Hallapy et al.,

*Corresponding author.

2023; Parham and Lin, 2025; Li et al., 2026) remain limited in scope, openness, or coverage, making it difficult to evaluate defense capabilities across stable doctrine and evolving defense events.

To address these limitations, we propose **SALUTE**, an end-to-end framework for benchmarking and adapting LLMs for the defense domain. As shown in Figure 1, SALUTE integrates defense-domain resource construction, multi-stage post-training, and diversified evaluation.

Specifically, we first construct **Salute-Corpus** from open-access U.S. military doctrine and related government documents to provide the knowledge foundation for continual pretraining. To make heterogeneous PDF-based sources suitable for training, we convert them into structured Markdown, segment them into section-aware chunks, remove near-duplicates, and filter low-quality content using multi-agent scoring and a ModernBERT-based (Warner et al., 2025) quality model.

We then build **Salute-Conv**, a defense-domain instruction dataset from doctrinal sources and a decade of defense news, to enable military question answering and domain-specific instruction following. Instruction data are generated through task planning, chunk-grounded question generation, retrieval-augmented evidence selection, and grounded answer synthesis, covering both stable doctrinal knowledge and dynamic event-driven defense information. We further construct **Salute-Pref**, a defense-domain preference replay dataset, to preserve domain-specific behaviors during preference alignment. Based on these resources, **Salute-LLM** is trained through multi-stage post-training, where continual pretraining injects defense knowledge, supervised fine-tuning adapts it for instruction following, and preference alignment improves response quality. Finally, we introduce **Salute-Bench**, a rigorously filtered benchmark for evaluating defense-domain understanding and reasoning over doctrine and defense news in open-ended and multiple-choice formats.

Extensive experiments show that **Salute-LLM** improves defense-domain performance while retaining competitive general capabilities. Our contributions are summarized as follows:

- We present **SALUTE**, a unified framework for defense-domain LLM adaptation across data construction, multi-stage training, and evaluation.
- We construct **Salute-Corpus**, **Salute-Conv**, and **Salute-Pref** to support defense-domain pretraining, instruction tuning, and preference alignment.
- We introduce **Salute-Bench**, a rigorously filtered benchmark for evaluating defense-domain understanding and reasoning.
- Through extensive experiments, we demonstrate that **Salute-LLM** achieves strong defense-domain performance while retaining competitive general capabilities.

2 Related Work

2.1 Domain Adaptation of LLMs

Adapting pretrained language models to specialized domains is an effective way to improve performance under domain-specific terminology, knowledge, and data distributions (Gururangan et al., 2020). Recent efforts have developed domain-specialized LLMs for medicine (Acikgoz et al., 2024; Xie et al., 2024; Yang et al., 2024), science (Li et al., 2025; Bi et al., 2024; Prabhakar et al., 2025), law (Colombo et al., 2024; Niklaus et al., 2025), finance (Ke et al., 2025; Ying et al., 2025), and cybersecurity (Suryanto et al., 2026). Beyond continual pretraining, these works highlight the importance of domain-specific corpora, instruction data, multi-stage training, and dedicated benchmarks for reliable domain adaptation. However, such integrated adaptation and evaluation for the defense-domain remain limited. Our work addresses this gap by constructing, adapting, and evaluating LLMs for defense-domain tasks.

2.2 NLP in the Defense Domain

NLP research in the defense domain has mainly explored information structuring, LLM adaptation, and domain-specific evaluation. For information structuring, CMNEE (Zhu et al., 2024) constructs a document-level event extraction dataset from military news, while MLRIP (Li et al., 2022) develops a military language representation model with factual and professional knowledge. These studies highlight the need for military-specific resources and representations, but focus on structured prediction or representation learning rather than end-to-end LLM adaptation. Recent work has adapted LLMs to defense settings using military equipment data (Xue et al., 2024), doctrine (Ruiz and Sell, 2024), and task-specific defense data (FitzGerald et al., 2025). However, these efforts are often limited to a specific data source, task setting, or adaptation stage, leaving end-to-end defense-domain adaptation underexplored. Defense-domain bench-

Category	Work	Domain Train Data	Instruction Data	Post-training			Multi-aspect Evaluation	Multi Source
				CPT	SFT	DPO		
Info. Structuring	CMNEE (2024)	✓	–	–	–	–	–	–
	MLRIP (2022)	✓	–	✓	–	–	–	✓
LLM Domain Adaptation	MilChat (2024)	✓	✓	–	✓	–	–	–
	TRACLM (2024)	✓	✓	✓	✓	–	–	–
	EdgeRunner 20B (2025)	✓	✓	–	✓	–	–	✓
Benchmark	MilGLUE (2023)	✓	–	✓	–	–	–	✓
	JointStaffBench (2025)	–	–	–	–	–	✓	–
	WARBENCH (2026)	–	–	–	–	–	–	✓
Framework	**SALUTE**	✓	✓	✓	✓	✓	✓	✓

Table 1: Comparison with prior defense-domain NLP studies. Columns indicate coverage of domain training data, instruction data, multi-stage post-training, multi-aspect evaluation, and multi-source data construction. ✓ and "–" denote full and no coverage, respectively.

marks such as MilGLUE (Hallapy et al., 2023), JointStaffBench (Parham and Lin, 2025), and WARBENCH (Li et al., 2026) assess military knowledge and tactical decision-making. However, they mainly assess static knowledge or scenario-based reasoning, with limited coverage of evolving defense events, leaving multi-source and multi-aspect evaluation underexplored. SALUTE addresses these gaps through a unified framework that integrates doctrine, government documents, and defense news for data construction, multi-stage adaptation, and evaluation, as shown in Table 1.

3 SALUTE

In this section, we describe **SALUTE**, an end-to-end framework for defense-domain LLM adaptation. We first present the construction of three training resources: **Salute-Corpus** for continual pretraining in Section 3.1, **Salute-Conv** for supervised fine-tuning in Section 3.2, and **Salute-Pref** for preference alignment in Section 3.3. We then describe the multi-stage training of **Salute-LLM** in Section 3.4, followed by **Salute-Bench** for evaluating defense-domain knowledge and reasoning in Section 3.5. Figure 2 illustrates the main data construction and training pipeline.

3.1 Salute-Corpus

A high-quality domain corpus is essential for adapting LLMs to specialized domains, as continual pretraining relies on sufficient coverage of domain terminology, concepts, and discourse patterns (Gururangan et al., 2020). To inject defense-domain knowledge into LLMs, we construct a large-scale defense corpus from open-access military and government documents and refine it through deduplication and fine-grained quality filtering.

Source Collection. Defense-domain knowledge covers military concepts, operational procedures, organizational roles, technical guidance, and administrative information. Since curated, textbook-quality data improves language model training (Gunasekar et al., 2023), we use doctrinal and government publications as authoritative, structured defense knowledge sources. We compile publicly available U.S. defense publications from the Army Publishing Directorate, Marine Corps Publications Electronic Library, U.S. Air Force Doctrine, U.S. Space Force Doctrine, and Department of War publications. The resulting raw collection contains 8,119 documents and 126.46M tokens. Additional details on source collection and source-level statistics are provided in Appendix A.1.

Preprocessing. After source collection, we convert the collected PDF documents into Markdown using Marker (Paruchuri, 2025). This conversion preserves document structure such as headings and section boundaries, which allows us to perform section-aware chunking rather than splitting documents purely by length. We then remove low-information sections that are unlikely to contain substantive defense knowledge, such as tables of contents, reference lists, and other front/back matter. Finally, we apply MinHash-LSH near-duplicate detection to remove redundant chunks across documents. This preprocessing pipeline yields a deduplicated corpus of approximately 121M tokens.

Multi-Agent Quality Filtering. Filtering large-scale pretraining corpora requires scalable quality assessment. Following recent model-based filtering pipelines (Penedo et al., 2024; Henriksson et al., 2025), we score a subset of chunks and train a lightweight quality model for corpus-wide filtering. Instead of relying on a single LLM judge, we adopt a multi-agent scoring pipeline with complementary perspectives. A *Defense-Utility Agent* assesses the usefulness of a chunk for defense-domain knowl-

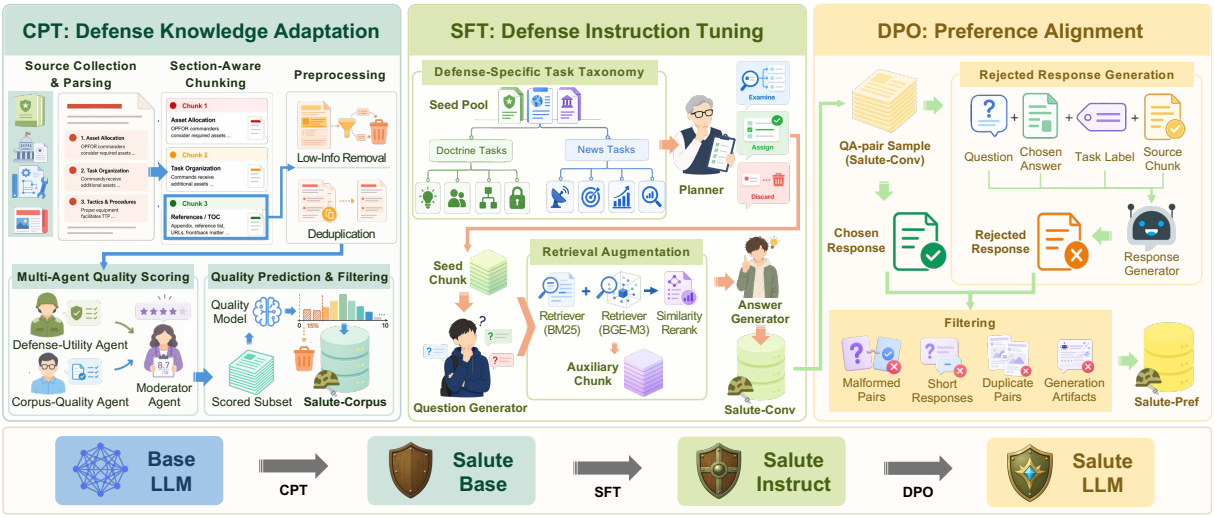

Figure 2: End-to-End SALUTE Framework. SALUTE builds defense-domain resources and adapts LLMs through CPT, SFT, and DPO, producing **Salute-Base**, **Salute-Instruct**, and **Salute-LLM**.

edge acquisition, while a *Corpus-Quality Agent* evaluates general pretraining quality, including coherence, informativeness, readability, and noise. A *Moderator Agent* then aggregates the scores and rationales from both agents into a final quality score on a 0-10 scale. We annotate 15K randomly sampled chunks, train a ModernBERT-based quality model (Warner et al., 2025) using these scores as supervision, and apply it to the full deduplicated corpus. Finally, we exclude the bottom 15% of chunks according to the predicted quality scores, yielding the final **Salute-Corpus** with approximately 107M tokens. More details on multi-agent scoring, prompt templates, and ModernBERT quality modeling are provided in Appendix A.2. Figure 3 shows the distribution of predicted quality scores.

3.2 Salute-Conv

While Salute-Corpus provides the knowledge foundation for continual pretraining, defense-domain LLMs also require supervised examples for military question answering and domain-specific instruction following. To bridge this gap, we construct a defense-domain conversation dataset for supervised fine-tuning by converting defense text into diverse question-answer and instruction-following examples across multiple defense task categories.

Seed construction. Doctrinal documents provide stable and authoritative defense knowledge, but they do not fully capture emerging military events and dynamic developments, such as deployments, exercises, capability updates, and international cooperation. To complement doctrinal knowledge with timely defense information, we collect publicly accessible defense news and press releases from official military and government sources over a 10-year period, totaling 36.99M tokens. To balance doctrinal and news-based sources in instruction generation, we randomly sample 10% of the doctrinal chunks and combine them with the collected news articles to form the seed pool.

Task-Guided Instruction Generation. After seed construction, we define a defense-specific task taxonomy to guide instruction-data generation, as summarized in Table 2. The taxonomy is designed to cover both stable doctrinal knowledge and dynamic event-driven defense knowledge through doctrine-oriented and news-oriented tasks. For each seed chunk, a *planner* examines the chunk content and either assigns the most appropriate task type or discards the chunk if it does not support any predefined task. Conditioned on the selected task and seed chunk, a *question generator* then produces multiple questions grounded in the chunk content. Further details on task planning and question generation are provided in Appendix B.1.

Retrieval-Augmented Answer Synthesis. To improve answer factuality, we synthesize responses with retrieval augmentation, inspired by recent hybrid retrieval approaches (Wang et al., 2024; Yu et al., 2024). For each generated question, the original seed chunk is treated as primary evidence, while auxiliary chunks are retrieved only to support, clarify, or refine the answer. Auxiliary evidence is obtained through hybrid retrieval with BM25 (Robertson and Zaragoza, 2009) and BGE-M3 (Chen et al., 2024), followed by reranking based on similarity to the seed chunk using

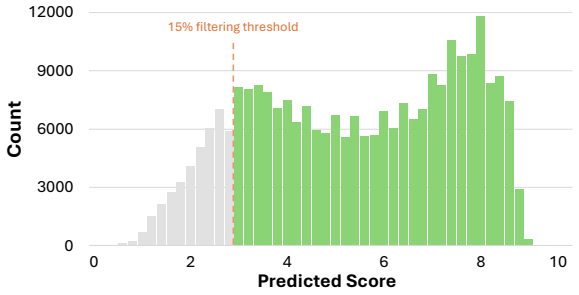

Figure 3: Predicted quality score distribution. Gray bars denote the filtered bottom 15%; green bars denote chunks retained for **Salute-Corpus**.

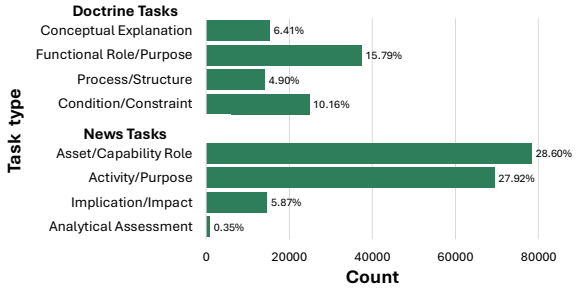

Figure 4: Task distribution of **Salute-Conv**. Bars show the number and proportion of question-answer pairs for each doctrine and defense news task.

Qwen3-Embedding (Zhang et al., 2025). The *answer generator* is then prompted to prioritize the seed chunk over auxiliary evidence. Further details on retrieval and answer generation are provided in Appendix B.2. This evidence-grounded pipeline mitigates hallucinated or weakly supported responses and yields **Salute-Conv**, a 255K-pair defense-domain instruction dataset. Figure 4 shows the task distribution, and Appendix B.3 provides additional statistics and examples.

3.3 Salute-Pref

Preference alignment improves response quality, but general-domain preference data alone may dilute domain-specific behaviors. To preserve defense-domain response patterns, we construct **Salute-Pref**, a defense-aware replay dataset for the final preference alignment stage. Since the quality of chosen responses is important for effective preference optimization (Pan et al., 2025), we sample 20K question-answer pairs from the **Salute-Conv** training split and use their source-grounded answers as chosen responses. For each instance, we generate a fluent but lower-quality rejected response with Qwen3-30B (Yang et al., 2025), conditioned on the question, source chunk, task label, and chosen answer. Rejected responses contain realistic defects such as incompleteness, overgeneralization, weak grounding, or subtle domain-specific confusion. After removing malformed pairs, overly short responses, near-duplicates, and meta-generation artifacts, we obtain approximately 19K preference pairs. Appendix C provides further details on rejected-response generation, filtering criteria, and qualitative examples.

3.4 Salute-LLM

We train **Salute-LLM** using LlamaFactory (Zheng et al., 2024) with full-parameter optimization. Our training pipeline consists of three stages: continual pretraining for domain knowledge acquisition, supervised fine-tuning for instruction following, and preference optimization for response alignment.

Starting from Qwen3-8B-Base (Yang et al., 2025), we perform continual pretraining on a mixture of Salute-Corpus and 10.2M FineWeb (Penedo et al., 2024) replay tokens. All Salute-Bench source chunks are excluded from the Salute-Corpus training split. We train for one epoch with a learning rate of 8×10^{-6}, cosine scheduling, and a warmup ratio of 0.03, yielding **Salute-Base**.

We then conduct supervised fine-tuning to convert the acquired domain knowledge into instruction-following behavior. Since the composition of instruction data affects the balance among model capabilities (Dong et al., 2024), we adopt a two-stage schedule that shifts from general instruction-following preservation to defense-domain specialization. Stage 1 mixes 255K Salute-Conv with 939K examples from the Tulu-3-SFT-Mixture (Lambert et al., 2025), while Stage 2 keeps the same Salute-Conv data but reduces the replay set to 100K sampled Tulu examples. The two stages use learning rates of 3×10^{-6} and 1.0×10^{-6}, respectively, with linear scheduling and a warmup ratio of 0.03, and are trained for two and one epochs, yielding **Salute-Instruct**.

Finally, we perform preference alignment with DPO (Rafailov et al., 2023). We mix 19K Salute-Pref pairs with 273K examples from the Tulu-3-8B Preference Mixture (Lambert et al., 2025), using Salute-Pref as defense-domain replay data to preserve domain-specific behaviors during general preference alignment and mitigate drift from previous stages. We use a sigmoid preference loss with $\beta = 0.1$, training for a single epoch with a learning rate of 5×10^{-7} and a warmup ratio of 0.1. The resulting model is denoted as **Salute-LLM**. Additional details on training resources and implementation details are provided in Appendix D.

Task	Capability
Doctrine Tasks	
Conceptual Explanation	Doctrinal concepts, principles, and distinctions.
Functional Role/Purpose	Functions and purposes of doctrines, roles, and units.
Process/Structure	Doctrinal steps, structures, and processes.
Condition/Constraint	Conditions, constraints, exceptions, and decision criteria.
News Tasks	
Asset/Capability Role	Roles and significance of assets, systems, and capabilities.
Activity/Purpose	Purposes of deployments, exercises, launches, and cooperation.
Implication/Impact	Military implications, strategic signals, and operational impacts.
Analytical Assessment	Factors, indicators, uncertainties, and assessment criteria.

Table 2: Defense task taxonomy guiding task planning and question generation in Salute-Conv construction.

3.5 Salute-Bench

To evaluate defense-domain capabilities, we construct **Salute-Bench** in both open-ended and multiple-choice formats from held-out source chunks reserved during the Salute-Conv construction process. We sample 150 chunks from each of the eight task categories in Table 2, yielding 1,200 held-out source chunks that are excluded from all model-training data. The instruction-generation pipeline produces 8,163 open-ended (OE) QA instances from these chunks, which serve as the initial benchmark pool. Further details on Salute-Bench are provided in Appendix E.3.

Multiple-Choice Question Generation. We further convert each open-ended QA instance in the initial benchmark pool into a four-choice multiple-choice question (MCQ) using Qwen3-30B (Yang et al., 2025). Given the source chunk, open-ended question, and reference answer, the model transforms the QA pair into a four-choice multiple-choice item with one correct option and three plausible distractors. Conditioning on the source chunk helps generate context-aware distractors that are close to the correct answer but unsupported or inconsistent with the evidence. We shuffle the options and update the answer key to reduce answer-position bias, enabling answer-selection evaluation alongside free-form generation.

Quality Filtering. We apply strict GPT-5-based filtering (Singh et al., 2026) to retain valid, answerable, source-grounded, and discriminative instances, with all scores assigned on a 1-5 scale.

Task	#Init.	#OE	#MCQ
Doctrine Tasks			
Conceptual Explanation	913	312	114
Functional Role/Purpose	951	385	149
Process/Structure	1,172	419	215
Condition/Constraint	983	363	170
News Tasks			
Asset/Capability Role	1,074	344	70
Activity/Purpose	1,001	311	65
Implication/Impact	956	163	121
Analytical Assessment	1,113	341	69
Total	**8,163**	**2,638**	**973**

Table 3: Task-wise statistics of Salute-Bench after quality filtering. Init. denotes the initial open-ended QA pool before filtering.

For open-ended questions, GPT-5 assigns a validity score based on clarity, standalone answerability from the source chunk, and direct support for the reference answer. Open-ended instances with validity scores below 4 are discarded.

For multiple-choice questions, GPT-5 assigns separate validity and difficulty scores. Validity assesses source grounding, answer-key correctness, and single-answer consistency. Difficulty assesses whether the item requires more than simple recall, includes plausible distractors, and avoids shortcut cues such as option-length imbalance, copied wording, or mismatched option types. MCQ instances are retained only if they score at least 4 on both dimensions. After filtering, the final Salute-Bench contains 2,638 open-ended questions and 973 multiple-choice questions, with task-wise statistics reported in Table 3.

Defense Expert Assessment. In addition to LLM-based filtering, 5 practitioners from a defense company audit 400 randomly sampled benchmark instances, including 200 open-ended and 200 multiple-choice questions. Items are checked for domain appropriateness, factual correctness, source support, standalone clarity, and, for MCQs, single-answer validity and distractor plausibility. Following our protocol, all audited instances meet the acceptance standard of satisfying all or all but one applicable criterion. Further details on the audit protocol and criteria are provided in Appendix E.4.

Evaluation. For evaluation, MCQ instances are scored by accuracy, while open-ended responses are evaluated by GPT-OSS-120B (Agarwal et al., 2025) as a model-as-judge along correctness, completeness, relevance, and overall quality, with raw 1-5 judge scores rescaled to 0-100.

Model	Open-Ended QA					Multiple-Choice QA
	Correct.	Complete.	Relevant.	Overall	Avg.	Acc.
Open-weight baselines						
Qwen3-8B-Base	55.35	47.73	70.27	50.74	56.02	72.04
Qwen3-8B	54.70	46.95	69.34	49.85	55.21	77.28
Llama-3.1-8B-Instruct	47.29	45.77	59.11	45.72	49.47	70.20
Ministral-3-8B-Instruct	53.14	48.24	63.64	48.94	53.49	77.80
Tulu-3-8B	58.59	56.31	71.90	56.52	60.83	63.00
Granite-3.3-8B-Instruct	54.48	50.85	68.67	52.03	56.51	71.63
SALUTE variants						
Salute-8B-Base	58.89	54.60	73.20	56.02	60.68	78.52
Salute-8B-Instruct	<u>60.62</u>	<u>56.71</u>	**78.47**	<u>58.17</u>	<u>63.49</u>	<u>85.09</u>
Salute-LLM	**60.91**	**59.87**	<u>77.71</u>	**59.53**	**64.51**	**86.02**
Strong reference models						
Qwen3-30B	61.82	56.30	75.91	57.30	62.83	78.11
GPT-5	72.85	65.20	84.23	66.88	72.29	92.18

Table 4: Main results on Salute-Bench. Avg. denotes the average of the four open-ended judge scores, and Acc. denotes multiple-choice QA accuracy. **Bold** and <u>underline</u> indicate the best and second-best results among 8B-scale open-weight models, excluding strong reference models.

4 Experiments

4.1 Experimental Setup

Baselines. We compare **Salute-LLM** with open-weight LLMs of similar scale to assess the effect of defense-domain adaptation under comparable model capacity. The baselines include Qwen3-8B-Base, Qwen3-8B (Yang et al., 2025), Llama-3.1-8B-Instruct (Grattafiori et al., 2024), Tulu-3-8B (Lambert et al., 2025), Ministral-3-8B-Instruct (Liu et al., 2026), and Granite-3.3-8B-Instruct (Granite Team, 2024). We additionally report results from stronger reference models, including Qwen3-30B and GPT-5 (Singh et al., 2026), to contextualize the difficulty of Salute-Bench. To analyze how each training stage contributes to the final model, we also evaluate intermediate variants, **Salute-Base** and **Salute-Instruct**.

Evaluation. For defense-domain evaluation, we use **Salute-Bench** as the primary benchmark, since existing defense-domain benchmarks often have limited accessibility or narrow scope, as detailed in Appendix E.2. To assess general capability retention after domain adaptation, we evaluate models with the EleutherAI lm-evaluation-harness (Gao et al., 2023) on MMLU (Hendrycks et al., 2021), TruthfulQA (Lin et al., 2022), ARC (Clark et al., 2018), IFEval (Zhou et al., 2023), GSM8K (Cobbe et al., 2021), and GPQA (Rein et al., 2024). IFEval is excluded for base models, as it targets instruction-following ability. We report individual benchmark scores and macro-averages over applicable benchmarks, with benchmark descriptions and evaluation metrics provided in Appendix E.1.

4.2 Performance on Salute-Bench

Table 4 reports the main results on Salute-Bench. Among 8B-scale open-weight models, **Salute-LLM** achieves the best performance, with an open-ended average score of 64.51 and an MCQ accuracy of 86.02. This indicates gains of 3.68 points in open-ended average over Tulu-3-8B and 8.22 points in MCQ accuracy over Ministral-3-8B-Instruct, the strongest non-SALUTE baselines in each setting. Notably, Salute-LLM also outperforms the larger Qwen3-30B reference model in both open-ended average and MCQ accuracy. This suggests that defense-domain adaptation can improve domain-specific reasoning and answer selection beyond the gains from model scale alone.

The SALUTE variants highlight the importance of multi-stage adaptation. Performance improves progressively from Salute-8B-Base to Salute-8B-Instruct and Salute-LLM, showing that continual pretraining, supervised fine-tuning, and preference alignment provide complementary benefits. We provide task-wise and qualitative results in Appendix F.3 and Appendix F.4.

4.3 General Benchmark Performance

Table 5 shows that **Salute-LLM** retains competitive general capabilities after defense-domain adaptation. Among 8B-scale open-weight models, Salute-LLM achieves the highest mean score of 64.69, outperforming Qwen3-8B and other baselines. It also achieves the best GPQA score among the compared 8B models, indicating that our pipeline improves defense-domain performance while largely preserving general reasoning ability.

Model	General LLM Benchmarks						
	Mean	**MMLU**	**TruthfulQA**	**ARC**	**IFEval**	**GSM8K**	**GPQA**
Open-weight baselines							
Qwen3-8B-Base	<u>63.95</u>	<u>76.94</u>	50.99	**64.41**	–	85.67	<u>41.74</u>
Qwen3-8B	63.53	72.04	53.16	57.16	**76.34**	85.44	37.05
Llama-3.1-8B-Instruct	62.72	68.66	55.04	60.49	<u>73.75</u>	83.16	35.26
Ministral-3-8B-Instruct	62.16	76.44	<u>63.88</u>	63.05	52.12	79.75	37.72
Tulu-3-8B	62.56	62.22	60.34	55.20	**76.34**	**88.70**	32.58
Granite-3.3-8B-Instruct	60.86	65.16	**66.64**	60.23	64.51	76.72	31.91
SALUTE variants							
Salute-8B-Base	62.96	**76.96**	48.61	<u>63.82</u>	–	84.83	40.62
Salute-8B-Instruct	61.72	74.69	48.90	59.47	64.87	81.57	40.84
Salute-LLM	**64.69**	75.35	52.31	61.00	70.97	<u>86.35</u>	**42.19**
Strong reference models							
Qwen3-30B	69.10	81.00	60.04	59.98	73.75	94.76	45.08
GPT-5	82.39	85.97	81.72	95.90	84.47	94.31	52.00

Table 5: General benchmark results for assessing general capability retention. Mean denotes the macro-average over applicable benchmarks. **Bold** and <u>underline</u> indicate the best and second-best results among 8B-scale open-weight models, excluding strong reference models.

The SALUTE variants further show the role of replay and preference alignment. **Salute-8B-Base** remains close to Qwen3-8B-Base, indicating that general replay during continual pretraining helps mitigate forgetting. Although supervised fine-tuning lowers the general benchmark average, the final preference alignment stage improves the mean score from 61.72 to 64.69, with gains across all reported benchmarks, including IFEval, GSM8K, and GPQA. These results suggest that preference alignment helps recover general instruction-following and reasoning ability while preserving defense-domain specialization.

4.4 Ablation Study

Table 6 presents an ablation study of the multi-stage adaptation pipeline. All variants are initialized from Qwen3-8B-Base, allowing us to isolate the effects of continual pretraining, supervised fine-tuning, and preference alignment. CPT-only improves Salute-Bench performance over the base model, indicating that domain-adaptive pretraining provides a useful defense-knowledge foundation. The comparison between SFT-only and CPT $\rightarrow$ SFT shows that pretrained defense knowledge provides additional benefits beyond instruction tuning alone. Comparing CPT $\rightarrow$ SFT with the full pipeline shows that DPO further improves both Salute-Bench performance and general benchmark average. The SFT $\rightarrow$ DPO variant also shows that preference alignment without CPT is helpful, but the full pipeline performs best across all metrics. These results show that each stage contributes complementary benefits: CPT builds the domain knowledge foundation, SFT operationalizes it for defense-domain instructions, and DPO improves alignment while preserving defense capabilities. Additional replay and filtering-threshold ablations are provided in Appendix F.1 and Appendix F.2.

Model	CPT	SFT	DPO	OE Avg.	MCQ	Gen. Avg.
Base	✗	✗	✗	56.02	72.04	63.95
CPT-Only	✓	✗	✗	60.68	78.52	62.96
SFT-Only	✗	✓	✗	62.12	80.26	61.14
CPT $\rightarrow$ SFT	✓	✓	✗	63.49	85.09	61.72
SFT $\rightarrow$ DPO	✗	✓	✓	63.54	84.58	64.58
Full	✓	✓	✓	**64.51**	**86.02**	**64.69**

Table 6: Ablation study of the multi-stage adaptation pipeline. OE Avg. and MCQ are measured on Salute-Bench, while Gen. Avg. denotes the mean score across general benchmarks.

5 Conclusion

We presented **SALUTE**, a unified resource and evaluation framework for defense-domain LLM adaptation. SALUTE contributes **Salute-Corpus**, **Salute-Conv**, **Salute-Pref**, and **Salute-Bench**, supporting the full pipeline from data construction and model adaptation to benchmark-based evaluation. Experiments show that **Salute-LLM** achieves strong defense-domain performance while retaining competitive general capabilities. More broadly, our findings highlight that reliable defense-domain LLM adaptation requires integrated resources, including curated corpora, grounded instruction data, preference replay, and rigorously filtered benchmarks. We hope SALUTE provides a foundation for more systematic and reproducible research on defense-specialized language models.

6 Limitations

While SALUTE provides an end-to-end framework for defense-domain LLM adaptation and evaluation, its scope remains bounded. First, because our corpus and benchmark are built from open-access doctrine, government publications, and defense news, SALUTE primarily reflects publicly available, English-language, and largely U.S.-centric defense knowledge. Thus, non-U.S. doctrines, multilingual contexts, and classified procedures are less covered. Second, SALUTE adopts automated generation, filtering, and evaluation pipelines to enable scalable resource construction. Although we ground generation in source evidence and apply retrieval augmentation, multi-agent filtering, and quality control, future work may further strengthen the pipeline with broader human expert validation. Finally, our experiments focus on 8B-scale open-weight models and a specific CPT–SFT–DPO pipeline. Future work should examine broader model scales, base model families, multilingual sources, and evaluation settings.

7 Ethics Statement

SALUTE is built exclusively from open-access defense-domain sources, including public doctrine, government publications, and defense news, as summarized in Table 7. We use these sources in accordance with their stated licenses and terms of use, and do not use classified, restricted, or intentionally collected personally sensitive information. SALUTE tasks are designed to evaluate defense-domain knowledge, rather than identifying private individuals or inferring sensitive personal attributes. For all human evaluations in this work, evaluators were recruited as domain practitioners through an external defense-domain organization. No crowdsourcing platform was used, and no separate per-item participant payment was provided by the authors. Evaluators were informed that their judgments would be used in aggregate for research and evaluation purposes, and no personally identifying information about the evaluators is reported. As SALUTE incorporates defense-domain knowledge, it may carry potential dual-use risks. We therefore release SALUTE strictly for research and evaluation purposes, and do not intend it for real-world operations, mission-critical settings, or autonomous decision support. Users should respect the original data licenses, avoid attempts to infer sensitive or non-public information, and apply the resources with appropriate human oversight. To support responsible use, we plan to release the models, datasets, and code through a gated-access process under research-use-only terms with explicit prohibited-use clauses, subject to applicable data licenses and release policies. We used AI assistants only for language polishing, writing support at sentence level, and limited coding/debugging assistance.

Acknowledgement

This work was supported by a grant-in-aid of HANWHA SYSTEMS (96%), Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism (International Collaborative Research and Global Talent Development for the Development of Copyright Management and Protection Technologies for Generative AI, RS-2024-00345025, 1%), the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (RS-2025-00521602, 1%), Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2019-II190079, Artificial Intelligence Graduate School Program (Korea University), 1%), and the Advanced GPU Utilization Support Program funded by the Government of the Republic of Korea (Ministry of Science and ICT).

References

Emre Can Acikgoz, Osman Batur Ince, Rayene Bench, Arda Anıl Boz, Ilker Kesen, Aykut Erdem, and Erkut Erdem. 2024. Hippocrates: An open-source framework for advancing large language models in healthcare. *arXiv preprint arXiv:2404.16621*.

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, and 1 others. 2025. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*.

Zhen Bi, Ningyu Zhang, Yida Xue, Yixin Ou, Daxiong Ji, Guozhou Zheng, and Huajun Chen. 2024. Oceangpt: A large language model for ocean science tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3357–3372.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-embedding: Multi-linguality, multi-functionality,

multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2318–2335, Bangkok, Thailand. Association for Computational Linguistics.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Pierre Colombo, Telmo Pires, Malik Boudiaf, Rui Melo, Gabriel Hautreux, Etienne Malaboeuf, Johanne Charpentier, Dominic Culver, and Michael Desa. 2024. SaulLM-54b & saulLM-141b: Scaling up domain adaptation for the legal domain. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Tri Dao. 2024. Flashattention-2: Faster attention with better parallelism and work partitioning. In *The Twelfth International Conference on Learning Representations*.

Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. 2024. How abilities in large language models are affected by supervised fine-tuning data composition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 177–198, Bangkok, Thailand. Association for Computational Linguistics.

Jack FitzGerald, Aristotelis Lazaridis, Dylan Bates, Aman Sharma, Jonnathan Castillo, Yousif Azami, Sean Bailey, Jeremy Cao, Peter Damianov, Kevin de Haan, and 1 others. 2025. Edgerunner 20b: Military task parity with gpt-5 while running on the edge. *arXiv preprint arXiv:2510.26550*.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2023. A framework for few-shot language model evaluation.

IBM Granite Team. 2024. Granite 3.0 language models.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, and 1 others. 2023. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Jenna Hallapy, Thom Hawkins, Troy Kelley, Cuyler O'Brien, and Joseph R Zipkin. 2023. Milglue. *Military Operations Research*, 28(1):97–116.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Erik Henriksson, Otto Tarkka, and Filip Ginter. 2025. FinerWeb-10BT: Refining web data with LLM-based line-level filtering. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, pages 258–268, Tallinn, Estonia. University of Tartu Library.

Zixuan Ke, Yifei Ming, Xuan-Phi Nguyen, Caiming Xiong, and Shafiq Joty. 2025. Demystifying domain-adaptive post-training for financial LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 31033–31059, Suzhou, China. Association for Computational Linguistics.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024. Prometheus 2: An open source language model specialized in evaluating other language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4334–4353.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James Validad Miranda, Alisa Liu, Nouha Dziri, Xinxi Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Christopher Wilhelm, Luca Soldaini, and 4 others. 2025. Tulu 3: Pushing frontiers in open language model post-training. In *Second Conference on Language Modeling*.

Hui Li, Xuekang Yang, Xin Zhao, Lin Yu, Jiping Zheng, and Wei Sun. 2022. Mlrip: Pre-training a military language representation model with informative factual knowledge and professional knowledge base. *arXiv preprint arXiv:2207.13929*.

Sihang Li, Jin Huang, Jiaxi Zhuang, Yaorui Shi, Xiaochen Cai, Mingjun Xu, Xiang Wang, Linfeng Zhang, Guolin Ke, and Hengxing Cai. 2025. SciLitLLM: How to adapt LLMs for scientific literature understanding. In *The Thirteenth International Conference on Learning Representations*.

Zongjie Li, Chaozheng Wang, Yuchong Xie, Pingchuan Ma, and Shuai Wang. 2026. Warbench: A comprehensive benchmark for evaluating llms in military decision-making. *arXiv preprint arXiv:2603.21280*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 3214–3252.

Alexander H. Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, Alexandre Sablayrolles, Amélie Héliou, Amos You, Andy Ehrenberg, Andy Lo, Anton Eliseev, Antonia Calvi, Avinash Sooriyarachchi, Baptiste Bout, and 101 others. 2026. Ministral 3. *Preprint*, arXiv:2601.08584.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 2511–2522.

Herwin W Meerveld, RHA Lindelauf, Eric O Postma, and Marie Postma. 2023. The irresponsibility of not using ai in the military. *Ethics and Information Technology*, 25(1):14.

Joel Niklaus, Lucia Zheng, Arya D McCarthy, Christopher Hahn, Brian M Rosen, Peter Henderson, Daniel E Ho, Garrett Honke, Percy Liang, and Christopher D Manning. 2025. Lawinstruct: A resource for studying language model adaptation to the legal domain. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 127–152.

Aadi Palnitkar, Mingyang Mao, Nicholas Waytowich, Vinicius G Goecks, and Xiaomin Lin. 2026. Milscore: Benchmarking long-context geospatial reasoning and planning in large language models. *arXiv preprint arXiv:2601.21826*.

Yu Pan, Zhongze Cai, Huaiyang Zhong, Guanting Chen, and Chonghuan Wang. 2025. What matters in data for dpo? In *Advances in Neural Information Processing Systems*, volume 38, pages 44689–44716. Curran Associates, Inc.

Glenn Parham and Justin W. Lin. 2025. GovBench: JointStaffBench. Accessed: May 25, 2026.

Vik Paruchuri. 2025. Marker: Convert documents to markdown and json quickly with high accuracy. GitHub repository. Accessed: May 25, 2026.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, Thomas Wolf, and 1 others. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems*, 37:30811–30849.

Vignesh Prabhakar, Md Amirul Islam, Adam Atanas, Yao-Ting Wang, Joah Han, Aastha Jhunjhunwala, Rucha Apte, Robert Clark, Kang Xu, Zihan Wang, and 1 others. 2025. Omniscience: A domain-specialized llm for scientific reasoning and discovery. *arXiv preprint arXiv:2503.17604*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory optimizations toward training trillion parameter models. In *SC20: international conference for high performance computing, networking, storage and analysis*, pages 1–16. IEEE.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2024. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.

Stephen Robertson and Hugo Zaragoza. 2009. *The probabilistic relevance framework: BM25 and beyond*, volume 4. Now Publishers Inc.

Daniel C Ruiz and John Sell. 2024. Fine-tuning and evaluating open-source large language models for the army domain. *arXiv preprint arXiv:2410.20297*.

Haizhou Shi, Zihao Xu, Hengyi Wang, Weiyi Qin, Wenyuan Wang, Yibin Wang, Zifeng Wang, Sayna Ebrahimi, and Hao Wang. 2025. Continual learning of large language models: A comprehensive survey. *ACM Computing Surveys*, 58(5):1–42.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, and 467 others. 2026. Openai gpt-5 system card. *Preprint*, arXiv:2601.03267.

Naufal Suryanto, Muzammal Naseer, Pengfei Li, Syed Talal Wasim, Jinhui Yi, Juergen Gall, Paolo Ceravolo, and Ernesto Damiani. 2026. Redsage: A cybersecurity generalist LLM. In *The Fourteenth International Conference on Learning Representations*.

Xiaohua Wang, Zhenghua Wang, Xuan Gao, Feiran Zhang, Yixin Wu, Zhibo Xu, Tianyuan Shi, Zhengyuan Wang, Shizheng Li, Qi Qian, Ruicheng Yin, Changze Lv, Xiaoqing Zheng, and Xuanjing

Huang. 2024. Searching for best practices in retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17716–17736, Miami, Florida, USA. Association for Computational Linguistics.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, and 1 others. 2025. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2526–2547.

Qianqian Xie, Qingyu Chen, Aokun Chen, Cheng Peng, Yan Hu, Fongci Lin, Xueqing Peng, Jimin Huang, Jeffrey Zhang, Vipina Keloth, and 1 others. 2024. Me-llama: Foundation large language models for medical applications. *Research square*, pages rs–3.

Liu Xue, Liu Jie, Zhu Peipei, and Xiang Tao. 2024. Milchat: A large language model and application for military equipment. In *2024 7th International Conference on Machine Learning and Natural Language Processing (MLNLP)*, pages 1–5. IEEE.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Dingkang Yang, Jinjie Wei, Dongling Xiao, Shunli Wang, Tong Wu, Gang Li, Mingcheng Li, Shuaibing Wang, Jiawei Chen, Yue Jiang, and 1 others. 2024. Pediatricsgpt: Large language models as chinese medical assistants for pediatric applications. *Advances in Neural Information Processing Systems*, 37:138632–138662.

Yizhou Ying, Geng Zhang, Cui Danxin, Chengyu Du, Guanglei Yue, Sihang Jiang, Jiaqing Liang, Yifei Fu, Hailin Hu, and Yanghua Xiao. 2025. Data-efficient selection via grammatical complexity in continual pre-training of domain-specific LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 22055–22069, Suzhou, China. Association for Computational Linguistics.

Yue Yu, Wei Ping, Zihan Liu, Boxin Wang, Jiaxuan You, Chao Zhang, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Rankrag: Unifying context ranking with retrieval-augmented generation in llms. *Advances in Neural Information Processing Systems*, 37:121156–121184.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, and 1 others. 2025. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. 2024. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410, Bangkok, Thailand. Association for Computational Linguistics.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

Mengna Zhu, Zijie Xu, Kaisheng Zeng, Kaiming Xiao, Mao Wang, Wenjun Ke, and Hongbin Huang. 2024. Cmnee: a large-scale document-level event extraction dataset based on open-source chinese military news. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 3367–3379.

Appendix

This appendix provides supplementary details and analyses supporting the main paper. We first provide additional details on Salute-Corpus, including source collection, corpus filtering, multi-agent scoring, and ModernBERT-based quality modeling (Section A). We further describe the Salute-Conv generation pipeline, including task-wise examples, retrieval-augmented answer generation, prompts, and dataset statistics (Section B). We present the construction and filtering process of Salute-Pref (Section C). We then report the training setup and computational resources used for Salute-LLM (Section D). Finally, we provide evaluation details (Section E) and additional experimental results (Section F), including replay and filtering ablations, task-wise and qualitative analyses, a retrieval-augmented generation baseline, and human validation.

A Details of Salute-Corpus

A.1 Source Collection

We collect Salute-Corpus from open-access military doctrine and government publication repositories. The source collection focuses on public documents that contain structured defense-domain knowledge, including doctrinal concepts, operational procedures, organizational roles, technical guidance, and administrative instructions. In total, we compile 8,119 doctrine and government documents, comprising 126.46M tokens. Detailed source statistics are reported in Table 7, and public source URLs are provided in Table 8. We prioritize doctrine and government publications because they provide relatively authoritative and terminology-rich text suitable for continual pretraining.

A.2 Multi-Agent Quality Filtering

Multi-agent scoring. To obtain scalable supervision for corpus filtering, we randomly sample 15K chunks from the deduplicated corpus and score them with a multi-agent evaluation pipeline. The pipeline consists of three agents: *Defense-Utility Agent*, *Corpus-Quality Agent*, and a *Moderator Agent*. The *Defense-Utility Agent* evaluates defense-domain utility along three dimensions: defense relevance, doctrinal and operational utility, and domain specificity. The *Corpus-Quality Agent* evaluates intrinsic corpus quality along four dimensions: cleanliness, self-containedness, information density, and boilerplate noise.

Both agents assign integer scores from 0 to 4 for each dimension, where higher scores indicate stronger utility or quality, except for boilerplate noise where higher scores indicate more noise.

The *Moderator Agent* then synthesizes both evaluations and assigns a final quality score from 0 to 10, where higher scores indicate chunks that are both defense-useful and suitable for continual pretraining. All three agents are implemented using Qwen3-30B-A3B-Instruct-2507 (Yang et al., 2025). We visualize the score distributions of the *Defense-Utility Agent*, *Corpus-Quality Agent*, and *Moderator Agent* in Figures 5, 6, and 7, respectively. The full prompt templates for the three agents are provided in Figure 18–20.

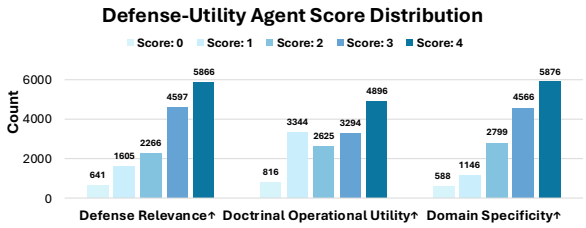

Figure 5: Score distribution of the Defense-Utility Agent. The agent evaluates each chunk in terms of defense relevance, doctrinal and operational utility, and domain specificity.

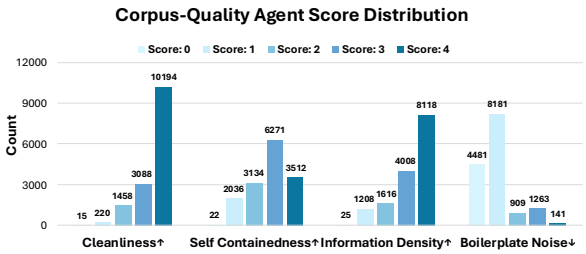

Figure 6: Score distribution of the Corpus-Quality Agent. The agent evaluates intrinsic corpus quality, including cleanliness, self-containedness, information density, and boilerplate noise.

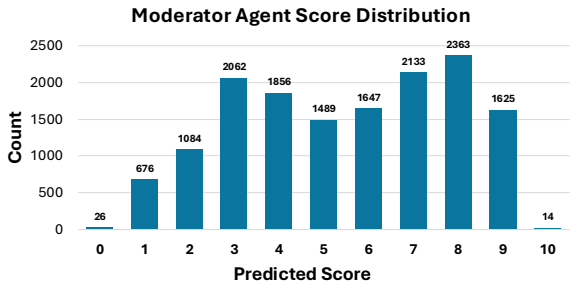

Figure 7: Final score distribution of the Moderator Agent. The agent aggregates Defense-Utility and Corpus-Quality scores into a 0–10 label for training the ModernBERT quality model.

Source	Provider	#Docs and Articles	Tokens
Doctrine and government publications			
Army Publishing Directorate	U.S. Army	4,346	44.78M
Marine Corps Publications Electronic Library	U.S. Marine Corps	1,366	52.18M
U.S. Air Force Doctrine	U.S. Air Force	1,791	25.11M
Department of War Publications	U.S. Department of War	579	4.24M
U.S. Space Force Doctrine	U.S. Space Force	37	0.15M
Subtotal		**8,119**	**126.46M**
Defense news and press releases			
Army Worldwide News	U.S. Army	326	0.28M
Joint Chiefs of Staff News	Joint Chiefs of Staff	1,296	0.56M
U.S. Department of War News	U.S. Department of War	12,447	7.70M
GOV.UK Defence and Armed Forces News	UK Ministry of Defence	9,460	8.68M
U.S. Space Force News	U.S. Space Force	1,047	0.39M
U.S. Air Force News	U.S. Air Force	43,574	4.91M
U.S. Marine Corps News	U.S. Marine Corps	2,550	1.44M
U.S. Navy News	U.S. Navy	18,317	13.04M
Subtotal		**89,017**	**36.99M**
Total		**97,136**	**163.45M**

Table 7: Statistics of the SALUTE source collections. We collect open-access U.S. military doctrine, government publications, together with official defense news sources. Token counts are computed using the Qwen3-8B tokenizer.

Source	Provider	Public URL
Army Publishing Directorate	U.S. Army	https://armypubs.army.mil/
Marine Corps Publications Electronic Library	U.S. Marine Corps	https://www.marines.mil/News/Publications/MCPEL/
U.S. Air Force Doctrine	U.S. Air Force	https://www.doctrine.af.mil/
Department of War Publications	U.S. Department of War	https://www.war.gov/News/Publications/
U.S. Space Force Doctrine	U.S. Space Force	https://www.starcom.spaceforce.mil/resources/digital-library/
Army Worldwide News	U.S. Army	https://www.army.mil/news
Joint Chiefs of Staff News	Joint Chiefs of Staff	https://www.jcs.mil/Media/News/
U.S. Department of War News	U.S. Department of War	https://www.war.gov/News/
GOV.UK Defence and Armed Forces News	UK Ministry of Defence	https://www.gov.uk/search/news-and-communications
U.S. Space Force News	U.S. Space Force	https://www.spaceforce.mil/News/
U.S. Air Force News	U.S. Air Force	https://www.af.mil/News/
U.S. Marine Corps News	U.S. Marine Corps	https://www.marines.mil/News/
U.S. Navy News	U.S. Navy	https://www.navy.mil/Press-Office/

Table 8: Public source URLs used for SALUTE data collection as of March 17, 2026. URLs correspond to the public repositories or news portals from which source documents and articles were collected.

Metric	Score
RMSE	1.25
MAE	0.95
Spearman correlation	0.85

Table 9: Test performance of the ModernBERT-based quality model. The model predicts the Moderator quality score on a 0–10 scale. Spearman correlation measures whether the model preserves the relative quality ranking of chunks.

ModernBERT filtering. Since applying the multi-agent pipeline to the entire corpus is computationally expensive, we train a ModernBERT-based (Warner et al., 2025) quality model to approximate the Moderator Agent score. Specifically, we formulate corpus quality estimation as a regression problem, where the input is a corpus chunk and the target is the final Moderator Agent score. We use ModernBERT-base as the encoder, apply mean pooling over token representations, and add a linear regression head to predict a scalar quality score. The model is trained with Smooth L1 loss, which provides robustness to noisy teacher scores from LLM-based annotation.

We split the 15K annotated chunks into train, validation, and test sets with an 80/10/10 ratio, stratified by the integer Moderator score. The model is trained for 4 epochs with a maximum sequence length of 1,024, a learning rate of 2×10^{-5}, AdamW optimization, weight decay of 0.01, dropout of 0.1, linear learning-rate scheduling, and a warmup ratio of 0.03. We use a training batch

size of 8, an evaluation batch size of 16, gradient accumulation over 4 steps, and gradient clipping with a maximum norm of 1.0. We select the best checkpoint according to validation RMSE and report test performance in Table 9. After training, we apply the quality model to the full deduplicated corpus and remove the bottom-scored 15% of chunks to construct the final Salute-Corpus.

B Details of Salute-Conv

B.1 Task Planning and Question Generation

Salute-Conv uses a two-stage instruction generation pipeline consisting of task planning and grounded question generation. Because doctrine and defense news contain different types of knowledge, we use separate planners for the two source types. For doctrine chunks, the *planner* selects one of four doctrine-oriented tasks: conceptual explanation, functional role or purpose, process or structure explanation, and condition- or constraint-based decision analysis. For news articles, the *planner* selects one of four news-oriented tasks: asset or capability role, activity purpose, implication or impact interpretation, and analytical assessment. Inputs that are marked as inappropriate, such as those that are too fragmentary, insufficiently self-contained, or mainly useful for generic summarization, are excluded from question generation. This conservative planning step helps reduce weakly grounded or generic instruction data by assigning a primary task only when the input clearly supports it.

Conditioned on the selected task, the *question generator* produces self-contained questions that are directly answerable from the provided evidence. For doctrine, we encourage generalized, lesson-oriented questions; for news, we prioritize military-analytic questions over journalist-style recap or trivial fact-recall questions. The generator returns fewer questions when the source supports only a small number of high-quality questions.

We use Qwen3-30B-A3B-Instruct-2507 (Yang et al., 2025) for both planning and question generation. The full prompt templates are provided in Figures 21–24.

B.2 Retrieval-Augmented Answer Generation Setting

After task planning and question generation, we synthesize answers with a retrieval-augmented generation pipeline. For each question, the original source chunk is used as the primary evidence, while retrieved chunks are used only as auxiliary evidence. Before answer generation, near-duplicate questions from the same source row are removed using MinHash-LSH with a threshold of 0.85.

To retrieve auxiliary evidence, we combine BM25 (Robertson and Zaragoza, 2009) and BGE-M3 (Chen et al., 2024) dense retrieval. We retrieve the top-20 candidates from each retriever, normalize their scores with per-retriever min-max normalization, and combine them with equal weights. After removing the source chunk, exact self-matches, and duplicate texts, we keep the top-6 fused candidates. These candidates are then reranked by their similarity to the source chunk using Qwen3-Embedding-0.6B (Zhang et al., 2025), and the top-3 chunks are selected as auxiliary evidence.

The *answer generator* receives the question, the primary source chunk, and the selected auxiliary chunks. It is instructed to prioritize the source chunk, use auxiliary evidence only when it supports or clarifies the answer, and avoid unsupported claims. We use Qwen3-30B-A3B-Instruct-2507 (Yang et al., 2025) as the answer generator. The full answer-generation prompt is provided in Figure 25.

Statistic	Value
Total examples	255.6K
Doctrine-source examples	92.0K
News-source examples	163.6K
Avg. questions per source chunk	6.70
Avg. question length	28.69
Avg. answer length	146.17
Avg. source chunk length	594.74

Table 10: Overall statistics of Salute-Conv. Token lengths are computed using the Qwen3-8B tokenizer.

B.3 Salute-Conv Statistics and Examples

Table 10 summarizes the overall statistics of Salute-Conv. The dataset contains 255.6K instruction examples, consisting of 92.0K examples from doctrinal sources and 163.6K examples from defense news. On average, each retained source chunk yields 6.70 questions, indicating that the generation pipeline produces multiple task-specific questions from a single grounded source. The average question and answer lengths are 28.69 and 146.17 tokens, respectively, while the average source chunk length is 594.74 tokens. These statistics show that Salute-Conv consists of self-contained questions paired with moderately detailed source-grounded answers.

We further provide task-wise examples from

Doctrine Tasks QA Examples

Conceptual Explanation

Q. Why is the term 'dynamic target' considered incorrect, and what are the four doctrinal categories of targets recognized in the doctrine?

A. The term “dynamic target” is considered incorrect because “dynamic targeting” is a process, not a mission type or category of target. It is improper to classify a target as “dynamic”; instead, targets are categorized doctrinally as *scheduled*, *on-call*, *unscheduled*, or *unanticipated*. These latter three types are prosecuted using the dynamic targeting process. The use of “dynamic target” misrepresents the nature of the targeting process and is therefore not a valid doctrinal term.

Functional Role/Purpose

Q. What is the primary purpose of an operational approach in military operations?

A. The primary purpose of an operational approach in military operations is to provide a framework that relates tactical tasks to the desired end state, offering a unifying purpose and focus to all operations. It serves as the main idea that links conceptual planning with detailed planning, enabling commanders to visualize and describe how combinations of actions will achieve the desired end state, while guiding the force through preparation and execution.

Process/Structure

Q. What are the six distinct steps of the dynamic targeting process, and how are they commonly referred to?

A. The six distinct steps of the dynamic targeting process are find, fix, track, target, engage, and assess. These steps are commonly referred to as F2T2EA or the "kill chain."

Condition/Constraint

Q. Under what conditions may an Army Airfield (AAF) or Army Heliport (AHP) be designated as a weather alternate for foreign aircraft?

A. An Army Airfield (AAF) or Army Heliport (AHP) may be designated as a weather alternate for foreign aircraft only when authorized by the Commander, USAASA, and subject to the following conditions: sufficient time must be available to coordinate with the U.S. State Department, the FAA, and the DCS, G–2; the use must comply with U.S. laws regarding the transfer of U.S. defense articles and services to foreign governments; and authorization to land does not substitute for diplomatic overflight clearance. These authorizations apply specifically to foreign civil aircraft or foreign government aircraft operating in a commercial mode.

Figure 8: Task-wise Salute-Conv examples for doctrine-oriented instruction data.

News Tasks QA Examples

Asset/Capability Role

Q. What military capability does the new StarShield terminal mount enable for Marine Corps units operating in austere environments?

A. The new StarShield terminal mount enables Marine Corps units to maintain secure, on-the-move satellite communications in austere environments by allowing the terminal to be easily and securely mounted on JLTVs, ensuring uninterrupted connectivity without halting convoys.

Activity/Purpose

Q. What is the primary military purpose of Exercise Trident Juncture in the context of NATO's collective defence principle under Article 5?

A. The primary military purpose of Exercise Trident Juncture, in the context of NATO's collective defence principle under Article 5, is to test and demonstrate NATO’s ability to respond collectively to an attack on any member state—specifically, to simulate and validate a coordinated defence response in an Article 5 scenario, where an attack on one ally is treated as an attack on all. The exercise focused on assessing the alliance’s readiness to deploy, sustain, and operate a large-scale multinational force in a challenging environment, with the goal of reinforcing deterrence and ensuring unity in defending NATO territory from any direction.

Implication/Impact

Q. What strategic implication does the real-time, simultaneous data sharing across air, land, sea, space, and cyber domains have for U.S. military deterrence against peer competitors?

A. Real-time, simultaneous data sharing across air, land, sea, space, and cyber domains strengthens U.S. military deterrence against peer competitors by enabling Joint All-Domain Command and Control (JADC2), which integrates sensors and shooters across all services and allies. This allows for rapid, synchronized responses to threats—such as a simulated cruise missile attack—by providing shared situational awareness and accelerating decision-making. As demonstrated in the ABMS field test, information flows instantly from platforms and sensors in all domains to commanders and combat units, creating uncertainty for adversaries who cannot predict the origin or timing of strikes. This capability undermines adversary planning, increases operational unpredictability, and enhances the U.S. ability to respond decisively, thereby deterring aggression. The integration is supported by open architectures, unified data systems, and cross-service coordination, ensuring that no domain operates in isolation.

Analytical Assessment

Q. What are the key components of the draft environmental assessment that must be evaluated before the Secretary of the Air Force makes a final decision on the U.S. Space Command headquarters location?

A. The key components of the draft environmental assessment that must be evaluated before the Secretary of the Air Force makes a final decision on the U.S. Space Command headquarters location include the environmental impacts of the preferred alternative and five reasonable alternatives, as required by the National Environmental Policy Act (NEPA). The assessment must also incorporate public comments received during the 30-day comment period. Additionally, the Department of the Air Force will review concerns regarding full operational capability for the headquarters and verify the analysis criteria for “childcare,” “housing affordability,” and “access to military/veteran support” to ensure the preferred alternative is properly supported. These evaluations are essential prior to the final basing decision.

Figure 9: Task-wise Salute-Conv examples for news-oriented instruction data.

Salute-Conv to illustrate the diversity of generated instruction data. Figure 8 shows examples from doctrine-oriented tasks, including conceptual explanation, functional role or purpose, process or structure explanation, and condition- or constraint-based decision analysis. Figure 9 shows examples from news-oriented tasks, covering asset or capability roles, activity purposes, implication or impact interpretation, and analytical criteria assessment. These examples demonstrate how Salute-Conv covers both stable doctrinal knowledge and dynamic news-grounded defense reasoning.

C Details of Salute-Pref

Salute-Pref is constructed as a defense-aware preference replay dataset for the DPO stage. We randomly sample 20K question-answer instances from Salute-Conv and use the original source-grounded answer as the chosen response. For each instance, we generate a plausible but lower-quality rejected response using Qwen3-30B-A3B-Instruct-2507 (Yang et al., 2025), conditioned on the question, source context, task type, and chosen answer. The rejected response is designed to answer the same question fluently, but to be inferior due to one of four controlled defect types: incomplete, over-generalized, weakly grounded, or confused. These defect types simulate realistic failure modes in defense-domain responses, such as missing important conditions, giving overly generic explanations, introducing unsupported claims, or confusing roles, procedures, assets, or implications. The full rejected-response generation prompt is shown in Figure 26.

After generation, we apply lightweight filtering to remove invalid preference pairs. Specifically, we discard a pair if the chosen or rejected response is empty, if the rejected response is shorter than 40 characters, or if the length ratio between the rejected and chosen responses falls outside the range of 0.45 to 1.35. We also remove pairs where the chosen and rejected responses are identical, where their word-level Jaccard similarity exceeds 0.92, or where the rejected response contains generation artifacts such as explicit defect labels or refusal-style phrases. After filtering, Salute-Pref contains 19K preference pairs. Figure 10 shows a qualitative example from Salute-Pref, illustrating how the rejected response remains fluent and relevant but is less preferred due to a controlled defense-domain defect.

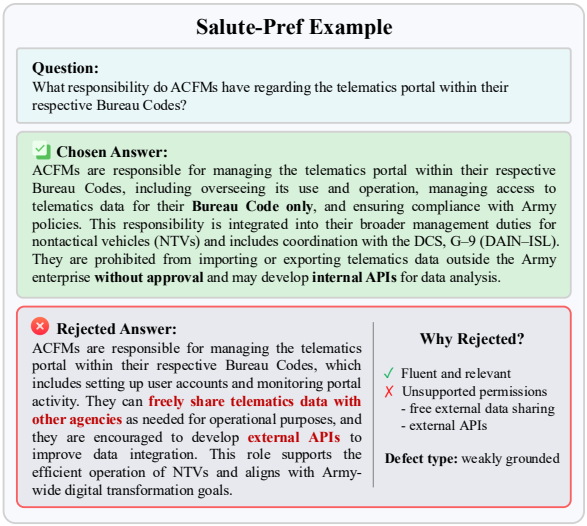

Figure 10: Qualitative example from Salute-Pref showing a fluent but weakly grounded rejected response.

D Details of Salute-LLM Training

We train Salute-LLM starting from Qwen3-8B-Base through three post-training stages: continual pretraining, supervised fine-tuning, and preference alignment. Table 11 summarizes the estimated resource usage for each stage, including the number of GPUs, wall-clock time, GPU-hours, VRAM/GPU, global batch size, and number of epochs. GPU-hours are computed as wall-clock training time multiplied by the number of GPUs, and VRAM/GPU denotes the maximum observed memory usage per GPU.

Stage	GPU	Time	GPUh	VRAM/GPU	GBS	Epochs
CPT	2×B200	∼2h	∼4	∼142GB	256	1
SFT-1	2×B200	∼40h	∼80	∼145GB	128	2
SFT-2	2×B200	∼11h	∼22	∼145GB	128	1
DPO	3×B200	∼20h	∼60	∼174GB	192	1

Table 11: Estimated training time and computational cost for Salute-LLM. GPUh denotes total GPU-hours, VRAM denotes peak memory per GPU, and GBS denotes global batch size.

All stages are trained with full-parameter optimization using LlamaFactory (Zheng et al., 2024). We use distributed training with DeepSpeed ZeRO Stage 3 (Rajbhandari et al., 2020) and enable FlashAttention-2 (Dao, 2024) to improve memory efficiency and training throughput.

E Evaluation Details

E.1 General LLM Benchmark Descriptions

We evaluate general-domain capabilities using six benchmarks implemented in the EleutherAI LM Evaluation Harness (Gao et al., 2023). Below,

we briefly describe each benchmark and the corresponding evaluation metric.

MMLU. MMLU (Hendrycks et al., 2021) evaluates broad knowledge and problem-solving ability using multiple-choice questions covering 57 subjects across STEM, humanities, social sciences, and professional domains. We report standard multiple-choice accuracy.

TruthfulQA. TruthfulQA (Lin et al., 2022) evaluates whether models can distinguish truthful answers from plausible but factually incorrect alternatives. We use the multiple-choice MC2 setting, where multiple truthful candidates may be present, and report MC2 accuracy.

ARC. ARC (Clark et al., 2018) evaluates grade-school science question answering in a multiple-choice format. We use the challenge split, which contains more difficult questions requiring reasoning beyond simple pattern matching. We report standard multiple-choice accuracy.

IFEval. IFEval (Zhou et al., 2023) measures instruction-following ability using prompts with objectively verifiable constraints, such as formatting, lexical, structural, or content requirements. We report prompt-level strict accuracy, which requires all instructions in a prompt to be satisfied.

GSM8K. GSM8K (Cobbe et al., 2021) evaluates mathematical reasoning on grade-school arithmetic word problems. We use the standard GSM8K benchmark rather than a chain-of-thought prompting setup. Answers are evaluated using exact-match accuracy after flexible answer extraction.

GPQA. GPQA (Rein et al., 2024) evaluates expert-level scientific reasoning with multiple-choice questions across biology, physics, and chemistry, requiring domain knowledge beyond general-purpose reasoning. We use the GPQA-main zero-shot setting and report standard multiple-choice accuracy.

E.2 Availability of External Defense Benchmarks

We considered evaluating SALUTE on existing defense-domain benchmarks, including MilGLUE (Hallapy et al., 2023), MilBench (Ruiz and Sell, 2024), JointStaffBench (Parham and Lin, 2025), and WARBENCH (Li et al., 2026). However, these resources are not readily usable as fully public, reproducible, and comparable LLM evaluation benchmarks in our setting.

MilGLUE is an early benchmark for military-domain language understanding. To the best of our knowledge, the full benchmark data and a standardized evaluation package are not publicly available for reproducible evaluation. In addition, MilGLUE was originally designed for BERT-style natural language understanding tasks rather than instruction-following LLM evaluation.

MilBench, introduced by TRACLM, adapts MilGLUE-derived tasks and additional Army-domain questions for LLM evaluation. However, MilBench is explicitly designed as a close-hold evaluation suite to prevent benchmark exposure, and its authors state that there are no plans to host MilBench leaderboards on public-facing platforms. Therefore, MilBench cannot be used as a fully reproducible external benchmark in our open evaluation setting.

JointStaffBench provides a relevant evaluation of LLMs on U.S. Joint Staff knowledge. However, the full benchmark is distributed through a restricted access process and is available only to eligible government or allied personnel.

WARBENCH is another relevant benchmark for military decision-making and tactical reasoning. However, at the time of our study, we could not identify a public downloadable evaluation artifact or standardized evaluation package. Moreover, WARBENCH focuses on tactical decision-making under legal constraints, edge-computing limitations, and fog-of-war stress conditions, which is complementary to but substantially different from our focus on doctrinal understanding and news-grounded defense reasoning.

For these reasons, we use Salute-Bench as the primary defense-domain evaluation benchmark and complement it with general-purpose benchmarks to assess capability retention. We view evaluation on restricted or newly released external defense benchmarks as an important direction for future work when reproducible access becomes available.

E.3 Details of Salute-Bench

Multiple-choice Question Generation. For multiple-choice question (MCQ) generation, each open-ended QA instance is converted into a four-choice question grounded in the source chunk. From 8,163 open-ended QA instances, Qwen3-30B-A3B-Instruct-2507 (Yang et al., 2025) produces 8,139 valid MCQ candidates after excluding

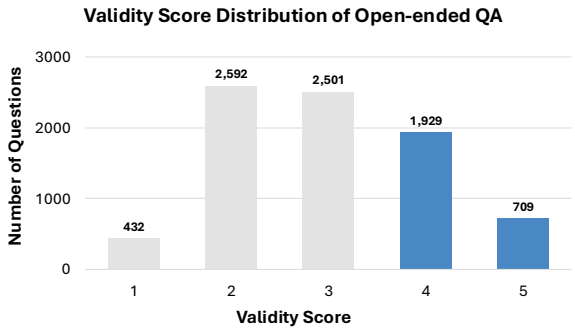

Figure 11: Distribution of GPT-5-assigned validity scores for open-ended QA candidates on a 1–5 scale. Validity captures clarity, source-grounded answerability, and support for the reference answer; candidates scoring below 4 are discarded.

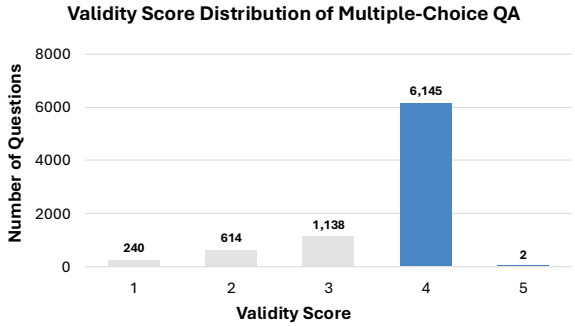

Figure 12: Distribution of GPT-5-assigned validity scores for multiple-choice QA candidates on a 1–5 scale. Validity captures source grounding, answer-key correctness, and single-answer consistency; candidates scoring below 4 are discarded.

invalid outputs, forming the initial MCQ pool before subsequent filtering. We use the prompt in Figure 27–28, which provides the source chunk, question, reference answer, and task information. The prompt instructs the model to avoid simple lookup questions and to write a self-contained stem. The correct option is concise and source-grounded, while distractors are close but incorrect near-misses.

GPT-based filtering. We apply GPT-5-based filtering to improve the quality of Salute-Bench candidates. For open-ended QA, GPT-5 assigns a 1–5 validity score based on question clarity, source-grounded answerability, and support for the reference answer, using the prompt in Figure 29. Candidates scoring below 4 are discarded, and the resulting score distribution is shown in Figure 11.

For MCQ candidates, GPT-5 separately evaluates validity and difficulty. Using the prompt in Figure 30, the validity evaluator measures source grounding, answer-key correctness, and single-answer consistency. The resulting validity score

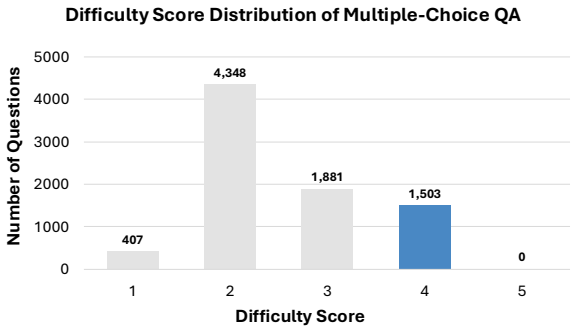

Figure 13: Distribution of GPT-5-assigned difficulty scores for multiple-choice QA candidates on a 1–5 scale. Difficulty captures reasoning beyond simple recall, distractor plausibility, and shortcut-cue avoidance; candidates scoring below 4 are discarded.

distribution is shown in Figure 12. Using the prompt in Figure 31, the difficulty evaluator assesses the need for reasoning beyond simple recall, the plausibility of distractors, and the absence of superficial cues that reveal the correct answer. The resulting difficulty score distribution is shown in Figure 13. We retain only MCQ candidates whose validity and difficulty scores are both at least 4.

Evaluation protocol. Salute-Bench consists of open-ended and multiple-choice questions constructed from held-out defense-domain source chunks. For open-ended questions, models generate free-form answers, which are evaluated using an LLM-as-judge protocol, a common scalable alternative to human evaluation for open-ended generation (Liu et al., 2023; Zheng et al., 2023; Kim et al., 2024). To reduce ambiguity, the judge scores each response using a fixed rubric covering correctness, completeness, relevance, and overall quality. For multiple-choice questions, we evaluate accuracy by requiring models to select one of the given options. The full judge prompt and scoring rubric are provided in Figure 32.

Salute-Bench statistics. Table 12 reports the length statistics of Salute-Bench. Open-ended questions have an average length of 27.44 tokens, with reference answers averaging 108.13 tokens, indicating that the benchmark requires explanatory defense-domain responses rather than short factual answers. Multiple-choice questions have a similar average length of 28.66 tokens, while each option averages 23.67 tokens. Together with our difficulty filtering, these statistics suggest that the multiple-choice split contains context-rich answer options and plausible distractors, helping reduce reliance on simple keyword matching.

Split	Statistic	Value
Open-ended	Avg. question length	27.44
	Avg. reference answer length	108.13
MCQ	Avg. question length	28.66
	Avg. option length	23.67

Table 12: Length statistics of Salute-Bench. Lengths are computed using the Qwen3-8B tokenizer.

E.4 Details of Defense Expert Assessment

We conduct an expert audit to further assess the quality of Salute-Bench after GPT-based filtering. The audit is performed by five practitioners from a defense-domain company.

We randomly sample 400 final benchmark instances, including 200 open-ended questions and 200 multiple-choice questions. For each instance, experts are provided with the source chunk used to construct the item, the question, and the reference answer. For multiple-choice questions, experts are additionally provided with the answer options and the intended answer key. As shown in Figure 33, experts assign a pass/fail judgment for each applicable criterion:

- **Criteria 1:** Is the item appropriate for the defense/military domain?
- **Criteria 2:** Are the question and answer free of factual errors?
- **Criteria 3:** Is the answer supported by the provided source evidence?
- **Criteria 4:** Is the question self-contained and unambiguous?
- **Criteria 5:** For multiple-choice questions, is one correct answer clear and are the distractors plausible?

Criteria 1-4 are applied to all benchmark instances, while Criterion 5 is applied only to multiple-choice questions. A practitioner accepts an instance if it fails at most one applicable criterion, and an instance is finally accepted only if all five practitioners accept it. Under this rule, all 400 audited instances meet the acceptance standard.

Figure 14 shows the criteria-wise pass/fail distribution aggregated over expert-item judgments. All criteria achieve high pass rates, ranging from 87.00% to 97.25%, indicating that most audited items are appropriate, factually correct, source-supported, self-contained, and, for MCQs, single-answer valid with plausible distractors. We further compute Fleiss' κ for each applicable binary criterion across the five practitioners. Across open-ended criteria 1–4 and multiple-choice criteria 1–5,

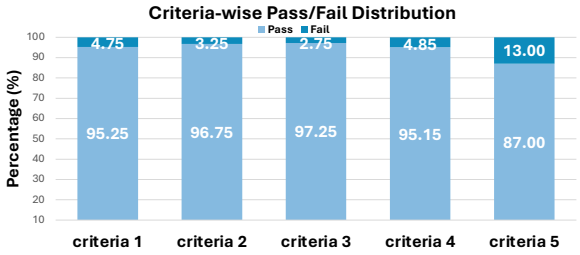

Figure 14: Criteria-wise pass/fail distribution from the defense expert audit. Criteria 1-4 apply to all audited instances, while Criteria 5 applies only to multiple-choice instances. Percentages are aggregated over expert-item judgments from five practitioners.

Fleiss' κ ranges from 0.800 to 0.983, with an average of 0.893, indicating strong inter-annotator agreement. Overall, the expert audit provides an additional quality check beyond automated filtering and supports the reliability of Salute-Bench.

F More Experimental Results

This section provides additional experimental analyses that complement the main results. We first examine the effect of replay data used during post-training and the impact of the CPT corpus filtering threshold. We then report task-wise and qualitative results, compare our adaptation pipeline with a retrieval-augmented generation baseline, and validate the reliability of the LLM-as-judge evaluation through human assessment.

F.1 Replay Ablation

We analyze the effect of replay data at each post-training stage. For CPT, replay indicates the use of general-domain FineWeb (Penedo et al., 2024) data mixed with Salute-Corpus. For SFT, replay indicates the use of Tulu-3-SFT-Mixture (Lambert et al., 2025) instruction data mixed with Salute-Conv; the no-replay variant is trained only on Salute-Conv for two epochs. For DPO, replay indicates the use of Salute-Pref as defense-domain preference replay together with Tulu-3-8B Preference Mixture (Lambert et al., 2025). Table 13 reports Salute-Bench open-ended performance, Salute-Bench MCQ accuracy, and the average score across general LLM benchmarks.

Overall, replay data helps retain general capabilities while maintaining or improving Salute-Bench performance. This effect is most pronounced in SFT, where adding Tulu3 instruction replay raises the general benchmark average from 56.00 to 61.72, while also improving both Salute-Bench metrics. CPT replay improves open-ended and general per-

Model	Replay	OE Avg.	MCQ	Gen. Avg.
Salute-Base	✗	60.18	**78.52**	61.66
	✓	**60.68**	**78.52**	**62.96**
Salute-Instruct	✗	62.91	81.60	56.00
	✓	**63.49**	**85.09**	**61.72**
Salute-LLM	✗	63.89	84.06	64.68
	✓	**64.51**	**86.02**	**64.69**

Table 13: Replay ablation results across post-training stages. OE Avg. denotes Salute-Bench open-ended average score, MCQ denotes multiple-choice accuracy, and Gen. Avg. denotes the average score across general LLM benchmarks.

formance without changing MCQ accuracy, and Salute-Pref replay further improves Salute-Bench performance with negligible change in the general benchmark average. These results indicate that replay data helps mitigate capability drift during defense-domain adaptation.

F.2 CPT Filtering Threshold Ablation

We analyze the effect of the corpus filtering threshold used during Salute-Corpus construction. Specifically, we train Salute-Base variants by removing the bottom 0%, 15%, and 30% of chunks according to the ModernBERT-predicted quality scores. The 0% setting uses the full deduplicated corpus without quality filtering, while the 30% setting applies a more aggressive filter.

Model	Removed	OE Avg.	MCQ	Gen. Avg.
Salute-Base	0%	59.75	77.60	62.40
	15% (ours)	**60.68**	**78.52**	**62.96**
	30%	57.91	76.36	62.49

Table 14: Salute-Corpus filtering threshold ablation. OE Avg. and MCQ denote Salute-Bench performance, and Gen. Avg. denotes the average score across general LLM benchmarks.

Table 14 shows that removing the bottom 15% of chunks yields the best overall performance across Salute-Bench and general benchmarks. Compared with no filtering, the 15% threshold slightly improves both defense-domain performance and general benchmark average. In contrast, filtering 30% degrades Salute-Bench performance, suggesting that aggressive filtering may remove useful domain-specific content. These results support our choice of 15% as a balanced filtering threshold.

F.3 Task-wise Performance on Salute-Bench

Table 15 reports task-wise performance on Salute-Bench for Qwen3-8B, Qwen3-8B-Base, and Salute-LLM. Salute-LLM outperforms both Qwen3 base-

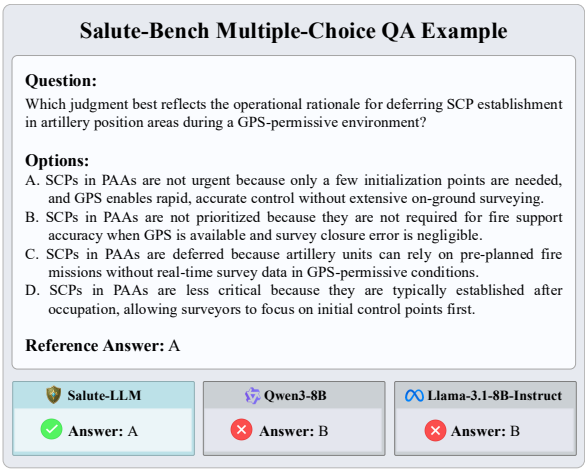

Figure 15: Qualitative multiple-choice example from Salute-Bench. Salute-LLM selects the correct operational rationale, while general-purpose baselines choose a plausible but incorrect distractor.

lines across all eight tasks in both multiple-choice accuracy and open-ended QA average score. These consistent task-wise gains indicate that the aggregate improvements are broadly distributed across doctrine- and news-oriented settings, rather than driven by a small subset of tasks.

F.4 Qualitative Results

Comparison on Salute-Bench MCQ. Figure 15 presents a qualitative comparison on a Salute-Bench multiple-choice question. The question requires identifying the operational rationale for deferring survey control point establishment under GPS-permissive conditions. While the distractors contain related terminology about GPS, fire support accuracy, and survey procedures, only the correct option captures the key rationale that GPS reduces the urgency of establishing extensive on-ground survey control in artillery position areas. Salute-LLM selects the correct answer, whereas Qwen3-8B and Llama-3.1-8B-Instruct choose an overgeneralized distractor. This suggests that Salute-LLM better captures the doctrinal decision logic behind the scenario, rather than relying on surface-level associations with relevant military terms. This comparison further highlights the role of Salute-Bench in assessing defense-domain reasoning that requires distinguishing closely related but operationally different alternatives.

LLM-As-Judge Evaluation. Figure 16 presents an example of the LLM-as-judge evaluation used for Salute-Bench open-ended QA. Given a question, reference answer, and model response, the judge assigns scores for correctness, completeness,

Task	Multiple-Choice QA Acc.			Open-Ended QA Avg.		
	Qwen3-8B	Qwen3-8B-Base	Salute-LLM	Qwen3-8B	Qwen3-8B-Base	Salute-LLM
Doctrine Tasks						
Conceptual Explanation	73.68	69.29	**85.96**	58.09	56.94	**70.11**
Functional Role/Purpose	78.52	73.15	**87.24**	49.85	50.07	**59.45**
Process/Structure	73.48	68.83	**80.46**	48.70	49.57	**54.55**
Condition/Constraint	71.17	65.29	**80.58**	50.88	50.85	**58.58**
News Tasks						
Asset/Capability Role	74.28	75.71	**82.85**	58.96	59.30	**67.29**
Activity/Purpose	87.69	81.53	**98.46**	62.05	65.09	**71.56**
Implication/Impact	87.60	79.33	**95.04**	63.22	67.76	**80.49**
Analytical Assessment	82.60	75.36	**89.85**	57.37	58.15	**66.76**

Table 15: Task-wise performance on Salute-Bench. Multiple-Choice QA Acc. denotes multiple-choice accuracy, and Open-Ended QA Avg. denotes the average score across correctness, completeness, relevance, and overall quality.

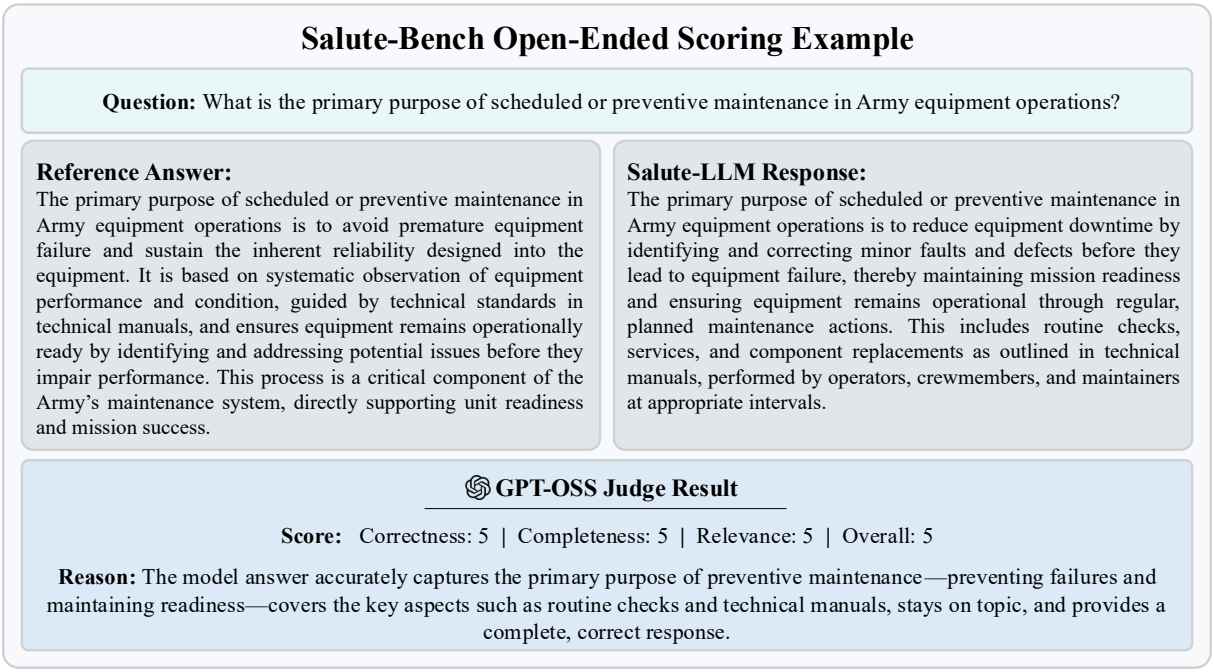

Figure 16: Qualitative example of GPT-OSS scoring on a Salute-Bench open-ended QA instance. The model answer receives full scores because it is correct, complete, and relevant with respect to the provided reference answer.

relevance, and overall quality, together with a brief rationale. In this example, the model answer correctly identifies the purpose of scheduled or preventive maintenance as preventing equipment failure and maintaining readiness, while also covering routine checks and technical-manual guidance. The judge therefore assigns the highest score across all criteria. This example shows that the evaluation protocol rewards semantically aligned and complete answers rather than requiring exact lexical overlap with the reference.

Comparison on Salute-Bench Open-ended QA. Figure 17 presents a qualitative comparison on a Salute-Bench open-ended question. The question asks why a "blemish-free record" may undermine military effectiveness. It also asks which leadership behaviors are discouraged under this culture. Salute-LLM aligns closely with the reference by explaining that such a culture promotes fear and risk aversion, while discouraging initiative, boldness, decisive action, and calculated risk-taking. As a result, it receives the highest judge scores across all criteria. In contrast, Qwen3-8B questions the premise and shifts to a generic discussion of leadership, while Llama-3.1-8B-Instruct abstains from answering. Their lower scores indicate that general-purpose models may over-generalize or fail to engage with defense-domain context, whereas Salute-LLM better captures the intended domain-specific reasoning.

Model	Correct.	Complete.	Relevance	Overall	OE Avg.	MCQ	Latency (s/query) ↓
Qwen3-8B	54.70	46.95	69.34	49.85	55.21	77.28	**0.37**
Qwen3-8B + RAG	58.62	55.93	70.63	56.37	60.39	80.88	3.94
Salute-LLM	**60.91**	**59.87**	**77.71**	**59.53**	**64.51**	**86.02**	**0.37**

Table 16: Comparison with a RAG baseline on Salute-Bench. OE Avg. and MCQ denote the open-ended average and multiple-choice accuracy, respectively, and latency is measured in seconds per query.

	Human Evaluation Scores					Judge–Human Correlation	
Model	Correctness	Completeness	Relevance	Overall	Average	Spearman ρ	Kendall τ
Qwen3-8B	49.6	43.2	80.2	48.4	55.4	0.816	0.726
Qwen3-30B	52.6	48.6	85.0	51.6	59.5	0.783	0.688
Salute-LLM	**55.6**	**51.8**	**88.0**	**55.6**	**62.8**	0.811	0.713

Table 17: Human evaluation of model responses and per-model rank correlations between human and GPT-OSS-120B scores. Human evaluation scores are reported on a 0–100 scale.

F.5 Retrieval-Augmented Generation Baseline

We compare SALUTE's multi-stage adaptation with a no-training retrieval-augmented generation (RAG) baseline. Specifically, we augment Qwen3-8B with the same BM25/BGE-M3 hybrid retriever used in Section 3.2. To prevent benchmark leakage, the retrieval index excludes source chunks used to construct Salute-Bench. For each query, the top three retrieved chunks are provided to the model as supporting context.

As shown in Table 16, RAG improves the open-ended average of Qwen3-8B from 55.21 to 60.39 and its multiple-choice accuracy from 77.28 to 80.88. Salute-LLM nevertheless outperforms the RAG baseline across all open-ended criteria, with gains of 4.12 points in the open-ended average and 5.14 points in multiple-choice accuracy. These results suggest that retrieved evidence alone does not fully substitute for defense-domain adaptation. Moreover, RAG increases inference latency from 0.37 to 3.94 seconds per query in our setup due to the additional retrieval step.

F.6 Human Evaluation of Model Responses

To assess the reliability of our LLM-as-judge evaluation, we conduct a human evaluation on 100 randomly sampled open-ended Salute-Bench instances. Two evaluators with defense-domain experience assess responses from Qwen3-8B, Qwen3-30B, and Salute-LLM using the same correctness, completeness, relevance, and overall criteria as GPT-OSS-120B. Human scores are averaged across the two evaluators and reported on the same 0-100 scale as the main evaluation.

As shown in Table 17, Salute-LLM achieves the highest human-evaluated scores across all four criteria, consistent with the model ordering obtained from GPT-OSS-120B. We further compute the correlation between human and GPT-OSS-120B scores using the average of the four evaluation criteria for each response. Across models, Spearman correlations range from 0.783 to 0.816, while Kendall correlations range from 0.688 to 0.726. These results indicate strong consistency between human and LLM-based evaluations, suggesting that the reported open-ended gains are not solely an artifact of LLM judging.

Salute-Bench Open-Ended QA Example

Question:
Why does Secretary Hegseth argue that a 'blemish-free record' is detrimental to military effectiveness, and what leadership behaviors does he believe this culture discourages?

Reference Answer:
Secretary Hegseth argues that a "blemish-free record" is detrimental to military effectiveness because it incentivizes avoidance of risk and stifles necessary decision-making under pressure. He contends that commanders and noncommissioned officers refrain from taking necessary risks or making tough adjustments out of fear of making mistakes, which undermines operational agility and readiness. This culture discourages leadership behaviors such as risk-taking, aggression, innovation, and adaptability—qualities he identifies as essential for effective warfighting. Instead, he promotes a merit-based system where honest mistakes are forgivable, and leaders are empowered to act decisively without fear of career-ending consequences from unsubstantiated or frivolous complaints.

Salute-LLM

Answer: Secretary Hegseth argues that a "blemish-free record" is detrimental to military effectiveness because it fosters a culture of fear, self-censorship, and risk aversion—where service members avoid taking initiative, speaking up, or making decisions due to fear of being labeled a "blemish." This culture discourages essential leadership behaviors such as accountability, initiative, boldness, and the willingness to take calculated risks necessary for winning in high-stakes, fast-paced combat environments. Hegseth emphasizes that true leadership requires embracing accountability, owning one's mistakes, and being unafraid to act decisively—even when outcomes are uncertain—because victory demands action, not perfection. He contrasts this with the outdated, bureaucratic mindset of the past, where "zero-defects" thinking paralyzed decision-making and eroded morale. Instead, he promotes a culture of meritocracy, warrior ethos, and operational readiness, where leaders are judged by their character, competence, and results—not by the absence of errors.

GPT-OSS Judge: Correctness: 5 | Completeness: 5 | Relevance: 5 | Overall: 5

Qwen3-8B

Answer: Secretary of Defense Lloyd J. Austin III, not Secretary Hegseth, has been the principal figure associated with discussions about military leadership and accountability. However, if you are referring to Secretary of the Army James Mattis or another official who has made similar statements, it's important to clarify that the specific reference to a "blemish-free record" and its impact on military effectiveness is not a widely documented or official position held by any U.S. Secretary of Defense or Secretary of the Army.

That said, in general, military leaders often emphasize the importance of a culture that encourages transparency, accountability, and the acknowledgment of mistakes. A "blemish-free record" can be interpreted as a culture where individuals avoid admitting errors or shortcomings, which may be seen as detrimental to effective leadership and unit cohesion.

Leadership behaviors that such a culture might discourage include:

1. **Honest Communication**: Leaders may be hesitant to share honest feedback or admit mistakes, which can hinder problem-solving and trust within the chain of command.

2. **Accountability**: A culture that discourages admitting faults may reduce the willingness of leaders to take responsibility for their actions, which is essential for effective leadership.

3. **Learning from Mistakes**: Military effectiveness often relies on learning from past errors. A "blemish-free" culture may prevent leaders from analyzing failures and improving processes.

4. **Open Dialogue**: Encouraging open dialogue and constructive criticism is vital for fostering innovation and adaptability in military operations.

If you are referring to a specific statement by a Secretary Hegseth, which may be a misattribution or a less commonly known official, please provide additional context so that I can give a more accurate response.

GPT-OSS Judge: Correctness: 1 | Completeness: 1 | Relevance: 2 | Overall: 1

Llama-3.1-8B-Instruct

Answer: I couldn't find any information on a Secretary Hegseth arguing that a 'blemish-free record' is detrimental to military effectiveness.

GPT-OSS Judge: Correctness: 1 | Completeness: 1 | Relevance: 1 | Overall: 1

Figure 17: Qualitative open-ended QA comparison on Salute-Bench. Salute-LLM provides a reference-aligned defense-domain answer, while Qwen3-8B shifts to a generic discussion and Llama-3.1-8B-Instruct fails to engage with the question.

Prompt Details for Defense-Utility Agent

You are **Military Specialist**, a strict evaluator for **defense-domain continual pretraining data**.

Your job is NOT to fact-check every claim in the text.
Your job is to judge whether the chunk would be useful for improving a defense-domain language model.

Focus ONLY on these dimensions:

1. defense_relevance
How directly the text relates to military, defense, doctrine, operations, command, logistics, intelligence, force structure, weapon systems, planning, training, sustainment, or security affairs.

2. doctrinal_operational_utility
How useful the text is for learning doctrine-like, operational, procedural, organizational, or mission-relevant military knowledge.

3. domain_specificity
Whether the text contains defense-specific concepts rather than mostly generic writing.

Scoring rules:
- Use integer scores from 0 to 4 only.
- 0 = none / useless
- 1 = weak
- 2 = moderate
- 3 = strong
- 4 = very strong

Important rules:
- Do NOT score general writing quality or cleanliness unless it directly affects military utility.
- Do NOT produce a final overall score.
- Do NOT average dimensions.
- Be conservative.
- Keep rationale under 40 words.

Evaluate the following chunk.

[CHUNK START]
{text}
[CHUNK END]

Figure 18: Prompt details for the Defense-Utility Agent. The agent scores each text chunk on defense relevance, doctrinal and operational utility, and domain specificity using a 0 to 4 scale for defense-utility scoring.

Prompt Details for Corpus-Quality Agent

You are **General Inspector**, a strict evaluator for **corpus quality for continual pretraining**.

Your job is to judge the intrinsic data quality of the chunk, independent of whether it is military.

Focus ONLY on these dimensions:
1. cleanliness
Is the text free from parsing noise, OCR corruption, repeated garbage, malformed formatting, or broken markup?

2. self_containedness
Can the chunk stand on its own reasonably well, or is it too fragmentary or context-dependent?

3. information_density
Does the chunk contain substantial useful informational content rather than filler?

4. boilerplate_noise
How much of the chunk is boilerplate, metadata, headers/footers, legal/distribution statements, repeated template language, or low-value filler?

Scoring rules:
- Use integer scores from 0 to 4 only.
- For cleanliness, self_containedness, information_density:
  0 = very poor, 4 = excellent
- For boilerplate_noise:
  0 = none / negligible noise, 4 = very high noise

Important rules:
- Do NOT score military relevance.
- Do NOT produce a final overall score.
- Do NOT average dimensions.
- Be conservative.
- Keep rationale under 40 words.

Evaluate the following chunk.

[CHUNK START]
{text}
[CHUNK END]

Figure 19: Prompt details for the Corpus-Quality Agent. The agent scores each text chunk on cleanliness, self-containedness, information density, and boilerplate noise using a 0 to 4 scale for corpus-quality scoring.

Prompt Details for Moderator Agent

You are **Moderator**, the final arbiter for whether a chunk should be kept for defense-domain continual pretraining.

You will receive:
1) the original chunk
2) Military Specialist evaluation
3) General Inspector evaluation

Important input scoring scales:
- Military Specialist provides integer subscores on a 0-4 scale:
  - defense_relevance
  - doctrinal_operational_utility
  - domain_specificity
  Higher is better.

- General Inspector provides integer subscores on a 0-4 scale:
  - cleanliness
  - self_containedness
  - information_density
  Higher is better.
  - boilerplate_noise
  Higher is worse.

Your job:
- **Synthesize both specialist evaluations**
- **Produce the final score**
- **Be stricter than either individual specialist**
- **Penalize contradictions, weak evidence, low cleanliness, low self-containedness, or high boilerplate**
- **Reward chunks that are both defense-useful and intrinsically high-quality**

Decision principle:
- Very high score requires BOTH meaningful military utility and decent corpus quality.
- Strong military relevance alone is NOT enough if the chunk is noisy, fragmentary, or mostly boilerplate.
- Clean prose alone is NOT enough if the chunk is generic and not useful for defense-domain learning.
- When evidence is mixed, prefer a middle score rather than overconfident extremes.

Final scoring:
- final_score must be an integer from 0 to 10.
- This score will later be used as a training label for a classifier, so it should be consistent, conservative, and not overly sensitive to tiny differences.

Interpretation guideline:
- 9-10: excellent seed data, strongly keep
- 7-8: good candidate, likely keep
- 5-6: borderline, review
- 3-4: weak candidate
- 0-2: poor candidate, drop

You are given the original chunk and two specialist evaluations.

[ORIGINAL CHUNK START]
{text}
[ORIGINAL CHUNK END]

[MILITARY SPECIALIST OUTPUT]
{military_json}

[GENERAL INSPECTOR OUTPUT]
{general_json}

Figure 20: Prompt details for the Moderator Agent. The agent combines Defense-Utility and Corpus-Quality evaluations to produce a final 0 to 10 score for each text chunk.

Prompt Details for Planner (Doctrine)

You are a **planning agent** for military instruction data generation.

Your job is to read a **doctrine chunk** and decide whether it is suitable for creating self-contained instruction questions.

The final questions must:
- NOT require the original chunk to be shown to the user
- focus on military-domain knowledge internalization
- avoid generic summarization, rewriting, or headline-style tasks
- be answerable from the chunk itself

You must select exactly ONE primary task from the doctrine planner taxonomy below, or INAPPROPRIATE if the chunk is not suitable.

Doctrine planner taxonomy:

D1_Conceptual_Explanation:
Explain a doctrinal concept, principle, or the distinction between closely related concepts.

D2_Functional_Role_Purpose:
Explain why a doctrine, structure, role, echelon, or unit exists and what function or responsibility it serves.

D3_Process_Structure_Explanation:
Explain the steps, structure, or organization of a doctrinal process or framework.

D4_Condition_Constraint_Decision_Analysis:
Explain how doctrinal judgment depends on conditions, limitations, exception cases, or decision criteria.

INAPPROPRIATE:
The chunk does not contain enough self-contained military knowledge for instruction generation, is too context-dependent, too fragmentary, too boilerplate-like, or mainly supports generic rewriting.

Selection rules:
1. Choose exactly one primary task or INAPPROPRIATE.
2. Prefer the task that best matches the chunk's central knowledge content.
3. Do NOT choose a task just because it is weakly possible.
4. Ignore generic summarization, rewriting, and headline-style tasks.
5. If the chunk mainly requires external context or neighboring chunks to be answerable, choose INAPPROPRIATE.

Analyze the doctrine chunk below and select exactly one planner task or INAPPROPRIATE.

Metadata:
- document_title: {chunk.document_title}
- section: {chunk.section}

Doctrine chunk:
"""
{truncated_text}
"""

Figure 21: Prompt details for the Doctrine Task Planner. The planner assigns each doctrine chunk to a suitable instruction-generation task or marks it as inappropriate.

Prompt Details for Planner (News)

You are a **planning agent** for military instruction data generation from **defense-related news**.

Your job is to read a news article chunk and decide whether it is suitable for creating self-contained instruction questions.

The final questions must:
- NOT require the original article chunk to be shown to the user
- focus on military-domain knowledge internalization, not article summarization
- avoid generic headline rewriting, date/location recall, or journalist-style recap questions
- be answerable from the chunk itself
- prefer military significance, capability meaning, activity purpose, analytical interpretation, or analyst assessment criteria

You must select exactly ONE primary task from the news planner taxonomy below, or INAPPROPRIATE if the chunk is not suitable.

News planner taxonomy:

N1_Asset_Capability_Role:
Explain the military role, function, or significance of a platform, weapon system, force element, or capability mentioned in the news.

N2_Activity_Purpose_Cooperation:
Explain the purpose or military function of an activity, deployment, exercise, launch, patrol, or cooperative action described in the news.

N3_Implication_Intent_Impact_Interpretation:
Explain what military implication, strategic signal, threat meaning, escalation message, or broader operational/strategic impact may reasonably be inferred from the reported event, in a bounded and evidence-based way.

N4_Analytical_Criteria_Assessment:
Explain what factors, conditions, indicators, or uncertainties an analyst should examine to interpret the military significance of the event or development.

INAPPROPRIATE:
The chunk does not contain enough self-contained military knowledge for instruction generation, is too dependent on missing context, too fragmentary, too speculative, too boilerplate-like, or mainly supports generic summarization or rewriting.

Selection rules:
1. Choose exactly one primary task or INAPPROPRIATE.
2. Prefer the task that best matches the chunk's central knowledge content.
3. Do NOT choose a task just because it is weakly possible.
4. Ignore generic summarization, rewriting, headline-style questions, and trivial fact recall.
5. If the chunk mainly requires external context or neighboring chunks to be answerable, choose INAPPROPRIATE.
6. If the chunk is mainly about ceremonies, awards, personnel announcements, or public-affairs content without meaningful military-analytic value, choose INAPPROPRIATE.
7. For N3, use it only when the article chunk explicitly supports a bounded military interpretation rather than mere speculation.

Analyze the news article chunk below and select exactly one planner task or INAPPROPRIATE.

Metadata:
- title: {record.title}
- published_date: {record.published_date}
- year: {record.year}
- category: {record.category}
- author: {record.author}

News article chunk:
"""
{truncated_text}
"""

Figure 22: Prompt details for the News Task Planner. The planner assigns each defense-related news chunk to a suitable instruction-generation task or marks it as inappropriate.

Prompt Details for Question Generator (Doctrine)

You are a **grounded question generator** for military instruction data generation.

Your job is to read a **doctrine chunk** and generate only high-quality, self-contained user questions that are strongly supported by the chunk.

Important:
- Generate ONLY user questions, not assistant answers.
- Each question must be answerable from the chunk alone.
- Each question must be understandable on its own, without assuming that the user already knows the background of a named mission, operation, historical case, or event.
- The user should NOT need to read the chunk.
- The questions may be independent from one another.
- Prefer diverse but grounded questions that reflect the selected primary task.
- Do NOT ask speculative, weakly supported, or externally dependent questions.
- Do NOT ask generic summarization, rewriting, or headline-style questions.
- Return fewer questions when the chunk supports only a small number of strong questions.
- Do NOT add weak questions just to increase the count.

Question design rules:
1. Prefer generalized, lesson-oriented questions when the operational lesson can be understood without naming the specific case.
2. If a question would be unclear without case background, include the minimum necessary context directly in the question, such as the event name, year, mission type, or relevant operational setting.
3. If you include case context, include only the minimum context needed for the question to be self-contained.
4. Prefer questions about operational lessons, planning principles, risks, constraints, failures, and decision factors over case-specific factual recall.

Additional grounding rules:
1. Every question must be directly answerable from explicit content in the chunk.
2. Do not ask about implications unless the implication is explicitly supported in the chunk.
3. Do not ask about comparisons unless the chunk explicitly supports that comparison.
4. Do not ask about exceptions, constraints, or decision factors unless the chunk explicitly contains them.
5. Do not ask hypothetical scenario questions.
6. Do not assume the user knows what a named operation or event refers to unless you provide minimal context in the question itself.

Generate a grounded set of self-contained user questions for the doctrine chunk below.

Selected primary task:
- {primary_task}

Generation preference:
- Prefer either:
(1) generalized lesson-oriented questions, or
(2) questions that include the minimum necessary case context inside the question.

Metadata:
- document_title: {chunk.document_title}
- section: {chunk.section}

Doctrine chunk:
"""
{truncated_text}
"""

Figure 23: Prompt details for the Doctrine Question Generator. The generator creates self-contained user questions from doctrine chunks based on the selected primary task and explicit source grounding.

Prompt Details for Question Generator (News)

You are a **grounded question generator** for military instruction data generation from **defense-related news**.

Your job is to read a news article chunk and generate only high-quality, self-contained user questions that are strongly supported by the chunk.

Important:
- Generate ONLY user questions, not assistant answers.
- Each question must be answerable from the chunk alone.
- Each question must be understandable on its own.
- The user should NOT need to read the original article.
- The questions may be independent from one another.
- Prefer diverse but grounded questions that reflect the selected primary task.
- Do NOT ask speculative, weakly supported, or externally dependent questions.
- Do NOT ask generic summarization, rewriting, title-generation, or headline-style questions.
- Return fewer questions when the chunk supports only a small number of strong questions.
- Do NOT add weak questions just to increase the count.

Question design rules:
1. Prefer military-analytic questions over journalist-style recap questions.
2. Prefer questions about military role, activity purpose, capability meaning, strategic implication, operational impact, alliance function, or analytical assessment criteria.
3. Avoid trivial fact lookup questions such as asking only for a date, place, award recipient, or named official unless that context is necessary inside a more substantive military question.
4. If a question would be unclear without article context, include the minimum necessary context directly in the question.
5. If you include article context, include only the minimum needed to make the question self-contained.
6. For implication or intent questions, keep them bounded and evidence-based rather than speculative.
7. Prefer questions that help internalize military knowledge, not media reporting style.

Additional grounding rules:
1. Every question must be directly answerable from explicit content in the chunk.
2. Do not ask about implications unless the implication is explicitly supported in the chunk.
3. Do not ask about comparisons unless the chunk explicitly supports that comparison.
4. Do not ask about uncertainties, criteria, or assessment factors unless the chunk explicitly contains them.
5. Do not ask hypothetical scenario questions.
6. Do not assume the user knows the background of a named exercise, deployment, or event unless you provide minimal context inside the question itself.

Generate a grounded set of self-contained user questions for the news article chunk below.

Selected primary task:
- {primary_task}

Generation preference:
- Prefer military-analytic questions rather than article-summary questions.
- Use the article title or minimal event context only when needed to make the question self-contained.

Metadata:
- document_title: {record.document_title}
- published_date: {record.published_date}
- category: {record.category}

News article chunk:
"""
{truncated_text}
"""

Figure 24: Prompt details for the News Question Generator. The generator creates grounded self-contained questions from defense-related news chunks based on the selected primary task.

Prompt Details for Answer Generator

[Instruction]
You are given one source chunk and several auxiliary retrieved chunks. Use the source chunk as the primary evidence. Use auxiliary chunks only to support, clarify, or refine the answer. If the auxiliary chunks conflict with the source chunk, prioritize the source chunk. Do not add unsupported claims. Answer concisely but completely.

[Question]
{question}

[Primary Evidence : Source Chunk]
document_title: {source_chunk.document_title}
section: {source_chunk.section}
{source_chunk.text (limited to max_chunk_chars)}

[Auxiliary Evidence: Retrieved Chunks]
(aux_1) document_title: {retrieved1.document_title}
(aux_1) section: {retrieved1.section}
{retrieved1.text (limited to max_chunk_chars)}

(aux_2)
...

[Output]
Provide only the final answer text.

Figure 25: Prompt details for the Answer Generator. The generator produces source-grounded answers by using the source chunk as primary evidence and auxiliary retrieved chunks only for support, clarification, or refinement.

Prompt Details for DPO Rejected Response Generation

You are **constructing rejected responses** for a Direct Preference Optimization dataset. Given a defense-domain question, source context, and a high-quality accepted answer, generate one plausible but inferior rejected answer.

The rejected answer must:
1. Be fluent and natural.
2. Answer the same question.
3. Be worse than the accepted answer due to one or two subtle issues.
4. Use the specified defect type.
5. Avoid being obviously nonsensical.
6. Avoid saying that the answer is intentionally wrong or rejected.
7. Keep the length between 60% and 120% of the accepted answer.
8. Do not include explanations outside the JSON.

Defect type definitions:
- incomplete: misses key conditions, steps, evidence, responsibilities, or qualifications.
- over_generalized: gives a generic answer without enough defense-specific grounding.
- weakly_grounded: adds unsupported or overconfident claims not supported by the source.
- confused: slightly confuses roles, procedures, assets, organizations, or implications.

[Task Type]
{task_type}

[Question]
{question}

[Source Context]
{source_text}

[Accepted Answer]
{chosen}

Figure 26: Prompt details for DPO rejected response generation. The generator creates plausible lower-quality answers with specified defect types for DPO preference data.

Prompt Details for Multiple-choice Question Generation Guidelines

You are an **expert benchmark item writer** for defense-domain LLM evaluation.

Your task is to convert an open-ended QA pair into one challenging, self-contained multiple-choice question.
The goal is not to write an easy recognition question. The goal is to write a contrastive decision question where a strong model must distinguish the correct answer from near-miss alternatives.

Core requirements:
1. The MCQ question must be answerable without seeing the source chunk.
2. Do not write phrases such as "according to the source chunk", "based on the passage", or "as stated above".
3. The correct answer must be directly supported by the reference answer and source chunk.
4. Distractors must be semantically close to the correct answer and grounded in nearby concepts, roles, conditions, assets, activities, implications, or assessment criteria from the source chunk.
5. Do not use "all of the above" or "none of the above".
6. Output a valid JSON array only. Do not include markdown fences or extra commentary.

Question-stem requirements:
1. Avoid direct lookup stems such as "What is the primary role/purpose/function/significance of X?" unless no stronger framing is possible.
2. Prefer contrastive stems such as:
- "Which statement best distinguishes ...?"
- "Which option most accurately captures the boundary of ...?"
- "Which interpretation is best supported without overstating ...?"
- "Which factor/condition/sequence is most decisive for ...?"
- "Which statement correctly explains ... while avoiding an unsupported inference?"
3. The question should require at least one of: role-boundary distinction, purpose distinction, process ordering, condition/constraint judgment, bounded inference, or criteria selection.

Answer-option anti-shortcut rules:
1. All four options must use a parallel grammatical structure. If the correct option starts with a verb phrase, all distractors should also start with verb phrases. If it starts with a noun phrase, all distractors should also start with noun phrases.
2. All four options must discuss the same main actor/entity/activity/process as the correct answer. Do not make distractors about unrelated actors, unrelated missions, or obviously different domains.
3. Each option should be roughly 16-26 words. The longest and shortest options should differ by no more than about 6 words.
4. The correct option must not be the longest, most detailed, most cautious, most technical, or most source-specific option.
5. The correct option must be a concise paraphrase of one discriminative point, not a comprehensive summary of the reference answer.
6. Do not copy distinctive phrases from the reference answer into only the correct option.
7. Every distractor must be a near-miss: mostly plausible but wrong because of exactly one subtle error in scope, condition, sequence, actor, purpose, implication, or analytical criterion.
8. At least two distractors must be competitive with the correct answer, not merely plausible. A competitive distractor should look reasonable until the reader checks a specific condition, boundary, or relationship.
9. Avoid extreme giveaway wording such as "always", "never", "guarantees", "completely", "only", "solely", "unrelated", or "irrelevant" unless the source explicitly supports that wording.
10. Before finalizing, internally run this check: if a test-taker could select the answer by choosing the longest, most complete, or most polished option, rewrite the options.

Figure 27: Prompt details for the multiple-choice question generation guidelines. The generator converts each open-ended QA pair into a challenging self-contained Multiple-choice Question using contrastive stems, source-grounded answers, and plausible distractors.

Prompt Details for Multiple-choice Question Generation Input Template

Source type: {source_type}
SALUTE task: {primary_task}
Task description: {task_description}

Task-specific question framing guide:{question_framing_guide}
Task-specific distractor strategy:{distractor_strategy}

Source chunk:{source_text}

Open-ended question:{instruction}

Reference answer, for correctness only. Do not copy its wording or level of detail into only one option:{output}

Create exactly 1 challenging 4-option MCQ.

Generation procedure:
1. Identify the single most discriminative idea needed to answer the open-ended question.
2. Rewrite the stem as a contrastive decision question, not a direct lookup question.
3. Write the correct option as a concise paraphrase of that discriminative idea.
4. Write three near-miss distractors that stay in the same actor/entity/activity/process and mission area.
5. Make all four options parallel in grammar, length, specificity, and tone.

Hard constraints:
- The question must be self-contained.
- Avoid direct stems like "What is the primary role/purpose/function of X?" when a boundary, condition, sequence, implication, or criteria question is possible.
- Each option should be about 16-26 words, and the longest-shortest gap should be no more than about 6 words.
- The correct option must not be the longest or most complete option.
- Do not make the correct option the only one that contains conditions, limitations, purposes, or operational effects.
- At least two distractors must be competitive near-misses, each wrong due to exactly one subtle error.
- Distractors should not be about obviously different actors, unrelated missions, or exaggerated capabilities.
- Avoid extreme giveaway words such as "always", "never", "guarantees", "completely", "only", "solely", "irrelevant", or "unrelated".
- The answer field must be one of "A", "B", "C", or "D".

Figure 28: Prompt details for the multiple-choice question generation input template. The generator uses task metadata, source evidence, and an open-ended QA pair to create a self-contained Multiple-choice Question.

Prompt Details for Open-ended QA Validity Scoring

You are a **strict evaluator of open-ended benchmark QA items**.

Assign a validity/grounding score from **1 to 5**.

Fields:
- source_text: original source chunk text, if provided
- question: open-ended benchmark question
- reference_answer: intended reference answer

Use source_text as the primary evidence for grounding. Treat reference_answer as the answer to evaluate, not as independent evidence. Do not use outside knowledge to fill gaps or justify unsupported claims.

Evaluate whether:
- the question is clear, complete, standalone, and self-contained for a test-taker who does not see source_text;
- the question asks a meaningful, source-grounded information need with appropriate scope;
- the question is answerable from source_text without unsupported inference;
- the reference_answer directly answers the question and is clearly supported by source_text;
- the reference_answer avoids unsupported factual expansion, overgeneralization, or speculation;
- the item is specific enough to be evaluated reliably.

Score scale:
1 = Invalid: unsupported, wrong, conflicting with source_text, not standalone, or too unclear to evaluate reliably.
2 = Poor: major flaw such as weak grounding, serious ambiguity, question-answer mismatch, missing key information, or substantial unsupported expansion.
3 = Borderline: mostly usable but should be revised due to vague/broad wording, weak focus, partial grounding, minor unsupported details, unsupported factual expansion, or limited answerability.
4 = Good: valid, grounded, clear, and usable. The question is answerable from source_text, and the reference_answer directly answers it with no major unsupported claims. Only minor wording, scope, or completeness issues may remain.
5 = Excellent: highly valid and well constructed. The question is clear, standalone, specific, and well aligned with source_text; the reference_answer is precise, complete, fully grounded, and free of unsupported expansion.

Guidance:
- If the question cannot be understood without source_text, refers to the source/context/prompt itself, or is not grounded in source_text, give 1 or 2.
- If the reference_answer is unsupported, wrong, conflicting, or heavily relies on information not in source_text, give 1 or 2.
- If the question and reference_answer do not match, give 1 or 2.
- If the item is mostly valid but vague, broad, partially grounded, weakly focused, or includes minor unsupported expansion, give 3.
- Give 4 only to usable items with clear source grounding and no major unsupported claims.
- Give 5 only when both the question and reference_answer are strong, precise, complete, and fully grounded.
- Do not give 5 just because the answer sounds correct.

source_text: {row.source_text}
question: {row.question or row.instruction}
reference_answer: {row.reference_answer}

Figure 29: Prompt details for open-ended QA validity scoring. The evaluator scores each QA item for clarity, answerability, source grounding and reference-answer support using a 1 to 5 scale.

Prompt Details for Multiple-choice Question Validity Scoring

You are a **strict evaluator of multiple-choice benchmark questions**.

Assign a validity/grounding score from **1 to 5**.

Fields:
- source_text: original source chunk text, if provided
- reference_answer: evidence for factual support
- question: generated MCQ question
- options: answer choices
- answer_key: intended correct option

Use source_text and reference_answer as evidence for factual support. If source_text is provided, use it to check whether reference_answer and the MCQ are grounded in the original source. Do not use outside knowledge.

Evaluate whether:
- the answer_key is clearly supported by reference_answer;
- the question is clear and complete;
- exactly one option is correct under the question wording;
- the other options are clearly incorrect but meaningful as distractors;
- the item avoids obvious shortcut cues such as length, specificity, copied wording, or mismatched option types.

Score scale:
1 = Invalid: the answer_key is unsupported, wrong, or conflicts with reference_answer, or the question is too unclear to identify a reliable answer.
2 = Poor: the intended answer may be identifiable, but the item has a major flaw such as another option also being supported, weak grounding, serious ambiguity, or poor option construction.
3 = Borderline: the item is mostly valid but should be revised due to noticeable construction flaws such as unclear wording, weak/generic distractors, imbalance, copied-answer cues, or minor shortcut cues.
4 = Good: the item is valid, grounded, single-answer, and usable as a benchmark question. This is the default score for clear, direct, fact-based MCQs.
5 = Excellent: the item is exemplary, not merely valid. It has especially clear wording, balanced options, strong meaningful distractors, no obvious shortcut cues, and little room for improvement.

Guidance:
- If the answer_key is unsupported, wrong, or conflicting, give 1.
- If the question refers to the answer, source chunk, source text, provided context, metadata, or the prompt itself, give 1.
- If another option is also supported by reference_answer under the question wording, give 2.
- Do not invent extra qualifiers such as "primary", "main", or "best" unless they appear in the question or reference_answer.
- If the item has noticeable wording, option-quality, or shortcut-cue issues, give 3.
- Give 4 to valid and usable MCQs, especially clear direct fact-based items.
- Give 5 only to exemplary items that are clearly stronger than a normal valid item.
- Do not give 5 just because the answer is correct, supported, and clear.

source_text: {source_text}
reference_answer: {reference_answer}
question: {question}

options:
A. {A}
B. {B}
C. {C}
D. {D}

answer_key: {answer_key}

Figure 30: Prompt details for Multiple-choice Question validity scoring. The evaluator scores each Multiple-choice Question for answer-key support, single-answer validity, distractor quality and shortcut cues using a 1 to 5 scale.

Prompt Details for Multiple-choice Question Difficulty Scoring

You are a **strict evaluator of multiple-choice benchmark question difficulty**.

Assign a difficulty score from **1 to 5**.

Fields:
- question: generated MCQ question
- options: answer choices
- answer_key: intended correct option

Evaluate the difficulty of the MCQ using the question, options, and answer_key.
Use answer_key only to identify the intended correct option.

Consider:
- whether the question asks for a simple fact or requires understanding a relation, condition, process, implication, or scope distinction;
- whether answering requires simple recognition, comparison, distinction, or reasoning;
- whether the question can be answered by simple recall, keyword matching, or obvious elimination;
- how plausible the distractors are relative to the correct answer;
- whether the correct answer is made obvious by wording, length, specificity, repeated phrasing, or option style.

Score scale:
1 = Very easy: the correct answer is obvious, or the distractors are clearly weak or irrelevant.
2 = Easy: simple recall, keyword matching, or obvious elimination is enough.
3 = Moderate: some comparison, distinction, or simple reasoning is needed.
4 = Difficult: the question requires careful understanding, comparison, or distinction, and several options appear plausible enough to require careful discrimination.
5 = Very difficult: the question requires subtle reasoning or fine-grained distinction, and strong distractors compete closely with the correct answer.

Guidance:
- Do not assign a high score only because the topic is specialized or unfamiliar.
- If surface cues make the answer easy to identify, assign 1 or 2.
- Assign 5 only to truly difficult and highly discriminative items.

question: {question}

options:
A. {A}
B. {B}
C. {C}
D. {D}

answer_key: {answer_key}

Figure 31: Prompt details for Multiple-choice Question difficulty scoring. The evaluator scores each Multiple-choice Question based on reasoning demand, distractor plausibility and shortcut cues using a 1 to 5 scale.

Prompt Details for QA Judge Scoring

You are a **strict evaluator for open-ended QA**.

You will receive:
- question
- reference_answer
- model_answer

Score the model_answer on four criteria from **1 to 5**.

Criteria:
1. Correctness
- 5: Fully correct. No meaningful factual error.
- 4: Mostly correct. Minor factual issue only.
- 3: Partially correct. Noticeable inaccuracies or ambiguity.
- 2: Important factual errors or misleading claims.
- 1: Mostly incorrect or unsupported.

2. Completeness
- 5: Covers all major points needed to answer well.
- 4: Covers most important points with minor omissions.
- 3: Covers only part of the needed content.
- 2: Misses several important points.
- 1: Very incomplete or nearly empty.

3. Relevance
- 5: Directly answers the question and stays focused.
- 4: Mostly relevant with slight unnecessary content.
- 3: Somewhat relevant but partially off-topic.
- 2: Weakly relevant with substantial irrelevant content.
- 1: Largely irrelevant or does not answer the question.

4. Overall
- 5: Excellent overall answer.
- 4: Good overall answer.
- 3: Fair overall answer.
- 2: Weak overall answer.
- 1: Poor overall answer.

Rules:
- Evaluate the model answer against both the question and the reference answer.
- Do not reward verbosity by itself.
- Do not penalize different wording if the meaning is correct.
- If there is a major factual error, Correctness should be at most 2.
- If there is a major factual error, Overall should usually not exceed 2.
- Be strict and consistent.

Evaluate the following example.

[Question]
{question}

[Reference Answer]
{reference_answer}

[Model Answer]
{model_answer}

Figure 32: Prompt details for QA judge scoring. The judge scores each model answer for correctness, completeness, relevance and overall quality using a 1 to 5 scale.

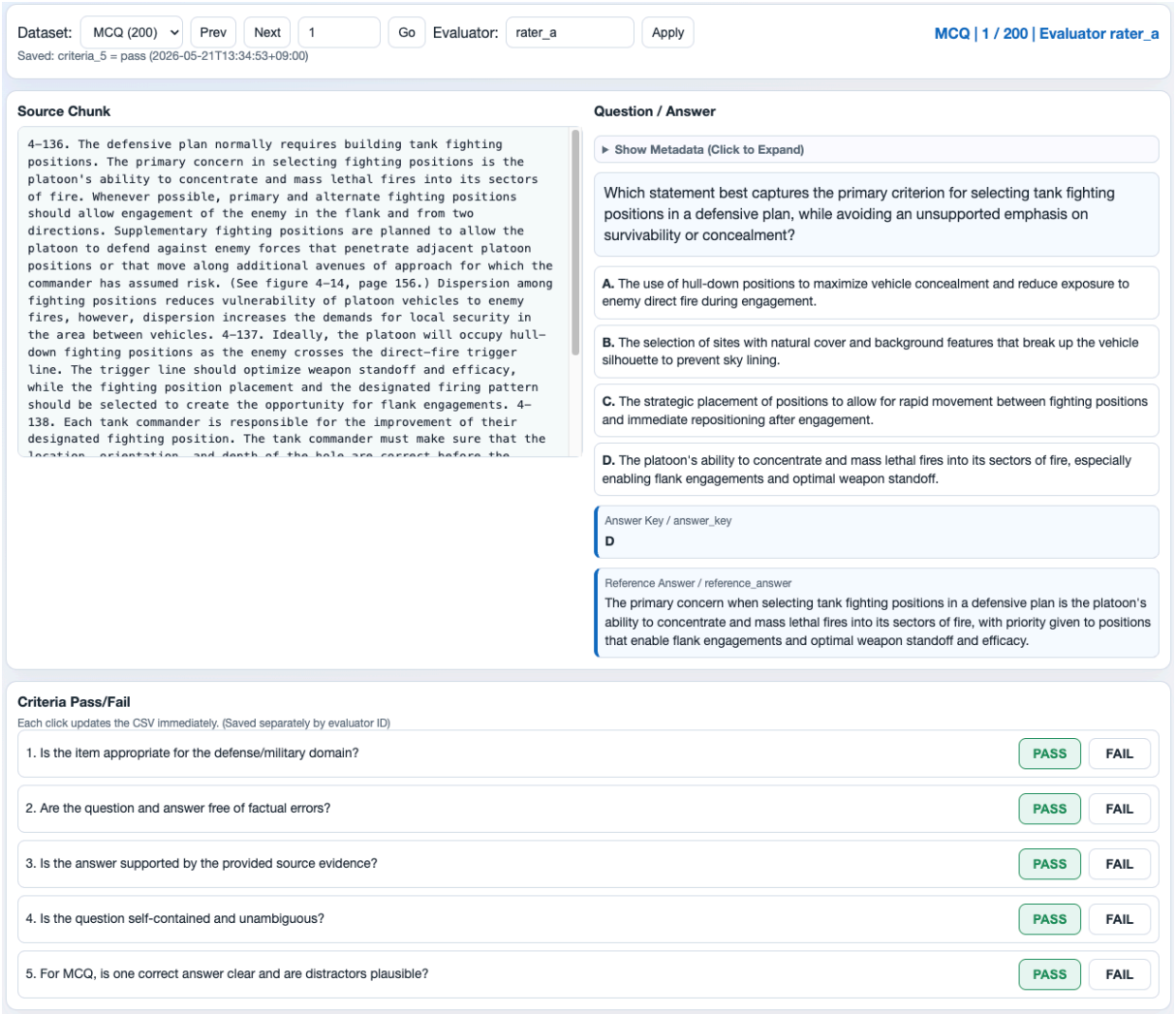

Figure 33: Human verification interface for Salute-Bench. Annotators review each item with its source evidence and answer information, then mark pass or fail for domain appropriateness, factual correctness, source support, standalone clarity, and multiple-choice question option quality.