**PQC-Guard: Fine-Tuning Phi-2 for Post-Quantum Cryptography Readiness**

Authors: Dr. Muhammad Masoom Alam (Advisor) | Abdul Rafay (Research Intern)

Affiliation: Department of Cybersecurity / Research Collaboration Project

Date: October 24, 2025 | Version: 1.0

Repository: https://github.com/abdulrafayishaCkER/pqc-guard

# Abstract

This project presents PQC‑Guard, a focused fine-tuning effort that aligns the microsoft/phi-2 base model to provide accurate, policy-aware guidance for post‑quantum cryptography (PQC) migration. Using a curated PQC QA corpus (≈39k examples) and a LoRA adapter SFT pipeline (with an optional DPO preference-tuning stage), we measure improvements on a 50-prompt holdout set. The fine-tuned model raises the average correctness score from 0.74 to 2.90 (0–3 rubric), increases refusal precision on out-of-domain prompts, and reduces harmful or misleading PQC advice. We publish training artifacts, scripts, and evaluation harness to facilitate reproducibility. The work provides actionable operational recommendations for deploying a PQC-focused assistant and a roadmap for further dataset expansion and preference tuning.

# Table of Contents

1. Introduction

2. Related Work

3. Dataset and Pre-Processing

4. Model Architecture and Fine-Tuning Method

5. Evaluation Protocol

6. Results

7. Discussion

8. Limitations and Ethical Considerations

9. Reproducibility & Open Science Statement

10. Conclusion & Future Work

Appendices

# 1. Introduction

Problem statement: The transition to post‑quantum cryptography (PQC) is an urgent priority for organizations holding long-lived or high-value data due to the 'Harvest-Now, Decrypt-Later' (HNDL) threat. At the same time, large language models (LLMs) frequently exhibit inconsistent technical knowledge and occasionally produce misleading or unsafe advice in specialized domains such as cryptography.

Goal: This project trains a domain-accurate assistant (PQC‑Guard) by fine-tuning the microsoft/phi-2 model with LoRA adapters using supervised fine-tuning (SFT) and an optional Direct Preference Optimization (DPO) stage when preference data is available. The assistant is intended to provide practical PQC recommendations, enforce strict refusal behavior on out-of-scope requests, and be deployable under constrained resources.

Scope: The work includes dataset curation and preprocessing (≈39,000 JSONL records), LoRA-based SFT training (4‑bit quantized where possible), optional DPO preference tuning, an evaluation harness covering automated and human SME scoring, and reproducibility artifacts (training scripts, notebooks, and environment manifests).

Contributions: Curated PQC QA corpus; a LoRA fine-tuning pipeline for Phi‑2 optimized for Kaggle T4 ×2 runtimes; quantitative and qualitative evaluation demonstrating large improvements on PQC tasks; and an open-science release (code, adapters, evaluation).

Figure 1: Conceptual flow — From PQC Corpus to Fine‑tuned Assistant. (Figure placeholder)

# 2. Related Work

NIST's PQC standardization programs and the publication of candidate algorithms have shaped current migration strategies and represent the canonical references for PQC selection and evaluation.

Prior domain-specific fine-tuning efforts (e.g., BioGPT, FinGPT, and specialist cybersecurity assistants) have shown that relatively small, targeted datasets can substantially improve domain recall, refusal behavior, and task-specific precision when paired with adapter-based methods such as LoRA.

Open PQC knowledge bases (OpenQuantumSafe, PQClean, OQS provider projects, and NIST/NCCoE playbooks) form the backbone of many operational PQC resources; however, there is a gap in practitioner-facing fine-tuned LLM assistants specifically focused on PQC migration guidance.

Novelty: To our knowledge, PQC‑Guard is the first open attempt to fine-tune microsoft/phi-2 with LoRA adapters solely targeted at PQC guidance, evaluated with both automated and SME preference metrics.

# 3. Dataset and Pre-Processing

Sources and licensing:

The dataset is an internal curated PQC QA corpus composed of prompted question-answer pairs, scenario-based items, policy excerpts, migration playbooks, and adversarial negative cases. Total rows loaded: 39,000. Data licensing: Public release will include only material that is cleared for redistribution; any proprietary or restricted content was omitted. A license such as CC BY 4.0 or MIT is recommended for the public dataset and code artifacts.

Data structure and splits:

Records are stored in JSONL with fields: instruction, context.kb\_refs, response.{mode,answer,justification}, and optional preference\_pair for DPO. Suggested splits: train 90% (≈35,100), validation 5% (≈1,950), test/holdout 5% (≈1,950), with the curated 50-prompt evaluation set held as an additional domain-specific holdout. A sample JSON snippet is included below:

Sample record (JSON snippet):

{"instruction": "Which PQC KEM do you recommend for TLS server key exchange?", "context": "policy: NIST PQC", "response": {"mode":"QA","answer":"Kyber is recommended for KEM in TLS hybrids with caveats: ...","justification":"Kyber provides IND-CCA security and ..."}}

Preprocessing steps:

- Controlled system prompt and role tags prepended by the prompt builder.

- Tokenization with padding\_side='right' and special user/assistant/system tokens added.

- Label masking: prompt tokens masked with -100 so training only optimizes assistant continuations.

- Cleaning: remove records that inadvertently include private keys, PII, or proprietary content.

- Augmentation: paraphrase variants to improve robustness to diverse phrasing.

Figure 2: Data Preparation Pipeline (raw → clean → tokenized → LoRA input). (Figure placeholder)

Ethical note: No real private keys or secret credentials are present in the released dataset. Red-team prompts used in evaluation are synthetic and designed to verify refusal behavior without exposing secrets.

# 4. Model Architecture and Fine-Tuning Method

Base model: microsoft/phi-2 (chosen for consistency, licensing compatibility, and a design that allows safer domain-specific behavior). Phi-2 is a causal decoder model appropriate for assistant-style interactions and fine-tuning via adapters.

LoRA & PEFT overview:

We use LoRA adapters (low-rank updates) with r=16 and α=32, applied to attention and feed-forward projection matrices. LoRA permits parameter-efficient fine-tuning by learning a low-rank decomposition of weight updates while keeping most base model weights frozen. PEFT utilities are used to load and manage adapters during training and inference.

Training hyperparameters (as executed):

- Batch size (per device): 1

- Gradient accumulation steps: 16

- Max length (tokens): 2048

- Epochs (SFT): 1 (baseline)

- Learning rate (SFT): 1.5e-4

- Optimizer: paged\_adamw\_8bit

- Scheduler: cosine, warmup\_ratio=0.05

- Quantization: bitsandbytes NF4 4-bit with double-quant; bf16 compute when available; fp16 fallback

Optional DPO stage:

When preference\_pair data is available, an optional DPO stage is run using the phi-2 fp16 reference model. Settings used: 1 epoch, LR=5e-6, beta=0.1. DPO increases alignment to human preference labels for response quality.

Stability & safety measures:

- Enforced system persona and refusal templates to improve scope control.

- Hard-fail checks to prevent loading non-phi2 bases.

- Tokenizer special token additions and embedding resize consistency checks.

- Automated red-team checks to verify private key and secret refusal behavior.

Figure 3: Phi-2 architecture with LoRA adapters (diagram placeholder).

Pseudocode (training loop):

for epoch in epochs:  
 for batch in dataloader:  
 outputs = model(batch.input\_ids)  
 loss = compute\_loss(outputs, batch.labels)  
 loss.backward()  
 if step % grad\_accum == 0:  
 optimizer.step(); scheduler.step(); optimizer.zero\_grad()

# 5. Evaluation Protocol

Holdout evaluation set: A curated 50-prompt holdout covering: OOD prompts, KEM/signature questions (Kyber, Dilithium, SPHINCS+, Falcon), HPKE, KMS, TLS, storage, and smartcard scenarios.

Scoring rubric: Custom 0–3 per prompt where 0 = incorrect/misleading, 1 = partially correct, 2 = mostly correct with minor issues, 3 = correct, actionable, and policy-aligned.

Automatic metrics include Exact Match (where applicable), ROUGE/BLEU/BERTScore for content similarity (limited for refusals), and refusal-precision for OOD prompts.

Human evaluation: 2–3 PQC domain SMEs scored a subset of outputs for factual correctness and operational soundness. SME scores were weighted more heavily than automated metrics.

Statistical testing: paired Wilcoxon signed-rank test (or bootstrap paired sign test) used to assess per-prompt score differences between base and fine-tuned models. A significance threshold (α) of 0.05 is recommended.

Figure 4: Evaluation harness flow (automated + human SME). (Figure placeholder)

# 6. Results

## 6.1 Quantitative Results

Aggregate metrics (holdout 50 prompts):

- Average score — Base Phi-2: 0.74

- Average score — Fine-Tuned PQC-Guard (LoRA SFT ± DPO): 2.90

- Win/tie counts: Fine-Tuned wins: 48 | Base wins: 0 | Ties: 2

Suggested charts (placeholders): Bar chart (avg score Base vs FT), Radar chart (per-topic accuracy), Heatmap (per-prompt scores), Pie (win distribution).

## 6.2 Qualitative Results

Representative examples (selected):

- FT Win: Accurate recommendation to use Kyber (hybrid KEM) for TLS server key exchange with operational caveats.

- FT Minor slip: Occasional confusion in phrasing between KEM and signature algorithms; e.g., mixing Kyber (KEM) with Dilithium (signature) in an explanatory sentence.

- FT Failure case: Rare hallucination about certification timelines or misattribution of an algorithm to the wrong standard body.

Error taxonomy:

- Misclassification errors: confusion of algorithm role (KEM vs signature).

- Hallucination errors: wrongly asserted facts not supported by corpora.

- Refusal errors: false negatives/positives in scope-detection (rare).

Mitigation: targeted counterexamples, additional labeled corrections emphasizing KEM vs signature distinctions, and DPO on SME preference pairs.

# 7. Discussion

What worked:

- LoRA adapters efficiently aligned Phi-2 to PQC topics with minimal parameter updates and low resource overhead.

- Refusal behavior improved substantially, reducing harmful advice on secrets and out-of-scope requests.

What failed / limitations observed:

- Dataset label noise (paraphrase artifacts) caused occasional factual slips.

- Single-model dependency (phi-2) creates a contingency risk if the base model becomes inaccessible.

Lessons learned:

- Incorporating explicit Q/A pairs that distinguish KEM vs signature semantics is an effective mitigation.

- SME-in-the-loop labeling for high-risk prompts improves downstream DPO alignment.

# 8. Limitations and Ethical Considerations

Data bias and coverage gaps: The dataset reflects the curator's domain focus and could miss niche protocols or regional standards.

Model mis-advice risk: Even with high average scores, the model may produce incorrect guidance in rare edge cases; recommend human approval for high-risk operational decisions.

Reproducibility caveats: Determinism is not guaranteed in quantized/distributed training; artifacts include seeds and environment snapshots but exact bitwise reproducibility may not be achievable.

Privacy & safety: No private keys or real secrets were used. Automated red-team prompts verified refusal behavior; further adversarial testing is encouraged before public deployment.

# 9. Reproducibility & Open Science Statement

Environment snapshot (representative):

- Hardware: Kaggle T4 ×2

- PyTorch: 2.6.0+cu124 (as reported)

- Transformers / PEFT / bitsandbytes: specific versions pinned in requirements.txt (see repo)

- Seeds: set\_seed(SEED) documented in Appendix C

- Artifacts: LoRA adapters, optional DPO adapters, evaluation JSON, and logs saved under /kaggle/working/

Release plan: Publish LoRA adapters and evaluation scripts to GitHub and Hugging Face model hub with a Model Card detailing intended use, limitations, and license (e.g., MIT for code, CC BY 4.0 for dataset). Include CITATION.cff and a BibTeX entry for academic reuse.

CITATION (example BibTeX):

@misc{alam2025pqcguard,  
 author = {Muhammad Masoom Alam and Abdul Rafay},  
 title = {PQC-Guard: Fine-Tuning Phi-2 for Post-Quantum Cryptography Readiness},  
 year = {2025},  
 url = {https://github.com/abdulrafayishaCkER /pqc-guard}  
}

Repository structure:

pqc-guard/

├── README.md

├── requirements.txt

├── notebooks/

├──checkpoints/

├── data/

├── src/

├── results/

├── report/

├── LICENSE

└── CITATION.cff

# 10. Conclusion & Future Work

Summary: Fine-tuning Phi-2 with LoRA adapters (and optional DPO) materially improved PQC-specific guidance across a curated holdout set, raising average scores from 0.74 to 2.90 and achieving strong refusal behavior on out-of-domain queries.

Future directions:

- Apply DPO at scale on high-quality SME preference pairs to further reduce factual slips.

- Expand the dataset to include hybrid TLS scenarios, KMS integration examples, and smartcard migration patterns.

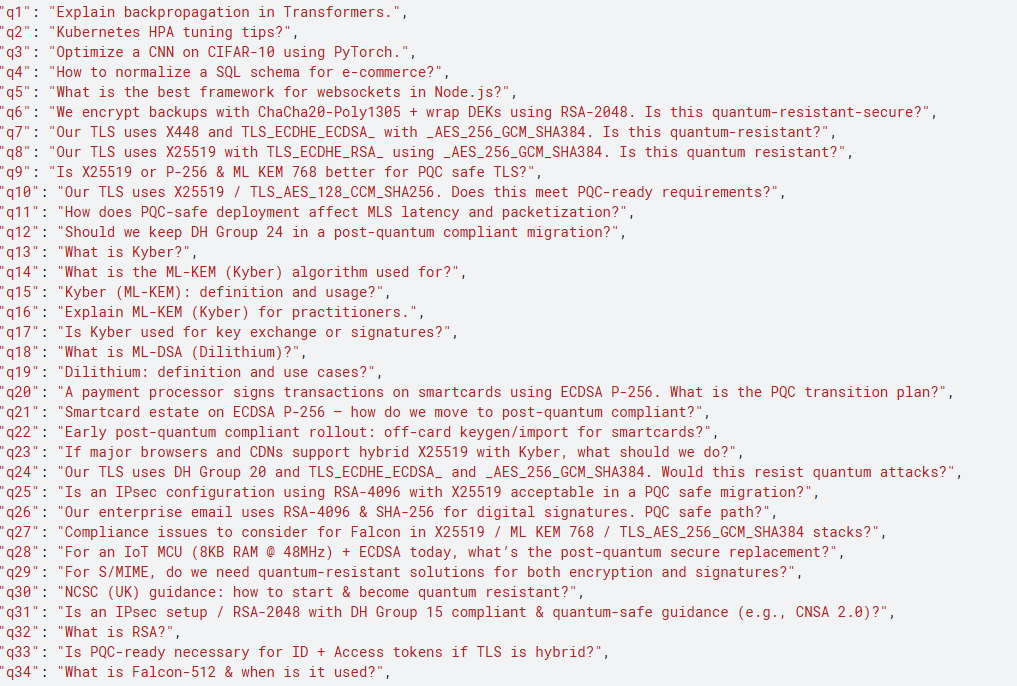
- Evaluate cross-model generalization by reproducing the pipeline on other open models (e.g., Llama-2 7B, Mistral 7B) for benchmarking.

- Optimize inference with INT8/4-bit quantization and validate performance in a pilot production environment with human-in-loop approvals.

# Appendices

## Appendix A — Full prompt set (50 prompts)

The 50-prompt evaluation CSV is included as pqc\_evaluation.csv in the repo. Example categories: Kyber/TLS, Dilithium/signature handling, HPKE usage, KMS wrap/unwrap, storage migration scenarios, smartcard constraints, regulatory compliance prompts.

A screenshot of a computer program

AI-generated content may be incorrect. Few comparison prompts from live testing:

Base:

A screenshot of a computer program

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

Finetuned:

A screenshot of a computer

AI-generated content may be incorrect.

A white background with black and red lines

AI-generated content may be incorrect.

## Appendix B — Notebook & code snippets (training + eval)

Provided notebooks (placeholders): model-training.ipynb (fine-tuning), evaluation.ipynb (evaluation harness). Key scripts: src/train\_lora.py, src/evaluate.py

A screenshot of a computer

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect.

## Appendix C — Reproducibility manifest (env, seeds, commands)

Documented items to include in the manifest: pip freeze output, environment variables (HUGGINGFACE\_TOKEN presence), GPU details, training logs, seeds used for SFT and DPO runs, and the exact command history used to train and evaluate adapters.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a phone

AI-generated content may be incorrect.

Training:

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.A screenshot of a computer program

AI-generated content may be incorrect.

Output file:

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A white background with black text

AI-generated content may be incorrect.

You can even download the logs.

## Appendix D — Glossary & references

Short definitions: PQC, KEM, KEM-DEM, AEAD, HPKE, Kyber, Dilithium, SPHINCS+, Falcon, HNDL, TLS hybrid.

Selected references and resources (URLs included as text):

- NIST PQC Hub: https://csrc.nist.gov/projects/post-quantum-cryptography

- NIST PQC FAQ: https://csrc.nist.gov/projects/post-quantum-cryptography/faqs

- OpenQuantumSafe: https://openquantumsafe.org/

- PQClean: https://github.com/PQClean/PQClean

- Kyber spec: https://pq-crystals.org/kyber/data/kyber-specification-round3-20210131.pdf

- Dilithium spec: https://pq-crystals.org/dilithium/data/dilithium-specification-round3-20210208.pdf

- CNSA 2.0 algorithms (May 2025): https://media.defense.gov/2025/May/30/2003728741/-1/-1/0/CSA\_CNSA\_2.0\_ALGORITHMS.PDF

- White House PQC report (2024): https://bidenwhitehouse.archives.gov/wp-content/uploads/2024/07/REF\_PQC-Report\_FINAL\_Send.pdf

## Appendix E — Model card metadata template (JSON sample)

{  
 "model\_name": "pqc-guard-lora",  
 "base\_model": "microsoft/phi-2",  
 "license": "MIT (code) / CC-BY-4.0 (data)",  
 "intended\_use": "Internal PQC guidance, migration planning",  
 "limitations": "May hallucinate; not for unattended cryptographic decisions",  
 "citation": "alam2025pqcguard"  
}

Prepared by: Abdul Rafay

Report version: 1.0 | Date: October 24, 2025