Kenya Clinical Reasoning Challenge: Solution Documentation

June 30, 2025

1 Introduction

This document presents an optimized solution for the **Kenya Clinical Reasoning Challenge**, hosted on Zindi, aimed at replicating the clinical reasoning of nurses in rural Kenyan healthcare settings. The challenge involves generating concise, clinically relevant assessments from 400 training and 100 test clinical vignettes, each combining nurse background (experience level, facility type) and complex medical scenarios across domains like maternal health and critical care. The solution employs a fine-tuned T5-base model, enhanced by data augmentation via back-translation, beam search reranking, and edge-optimized inference, achieving a public leaderboard ROUGE-L score of 0.4017. The entire pipeline runs in under 30 minutes, with training completed in 15 minutes on NVIDIA T4 GPUs, ensuring efficiency for edge device deployment. This documentation addresses the evaluation criteria: clarity (25%), insights (15%), implementability (25%), novelty (25%), and code quality (10%).

2 Problem Understanding

The Kenya Clinical Reasoning Challenge requires predicting clinician responses for 400 training and 100 test vignettes, reflecting real-world scenarios faced by nurses in resource-constrained Kenyan settings. Each vignette includes:

- **Patient Presentation**: Detailed medical scenarios across diverse domains.
- **Nurse Context**: Experience level, facility type, clinical panel, and nursing competency.
- **Constraints**: Small dataset (400 training samples), missing data (25% of 'Years of Experience'), class imbalances in clinical panel and nursing competency, and edge device requirements (<100ms inference per vignette, <1B parameters, <2GB RAM).

The evaluation metric is ROUGE-L, emphasizing precise text generation. Solutions must be fast, accurate, and deployable on edge devices like NVIDIA Jetson Nano, balancing clinical accuracy with resource limitations.

3 Solution Approach

3.1 Exploratory Data Analysis (EDA)

- **Insight**: 25% of 'Years of Experience' values were missing, and 'Clinical Panel' and 'Nursing Competency' had imbalances (some categories with ≤ 10 samples), risking poor generalization.
- **Action**: Created a `Length_Category` feature by stratifying response lengths into quartiles to analyze variability and guide training.
- **Visualization**: Plotted prompt and response length distributions, showing 55.8% of responses under 110 words, informing a `MAX_TARGET_LENGTH` of 128 tokens for concise outputs.

3.2 Data Preprocessing

- **Text Cleaning**: Implemented `preprocess_text` to lowercase text, remove punctuation, normalize medical measurements (e.g., "500mg" to "500 mg"), and collapse spaces, ensuring dataset consistency.
- **Prompt Enhancement**: Developed `create_prompt` to enrich prompts with nurse context (clinical panel, nursing competency, experience) and patient demographics (age, gender) extracted via regex, improving input informativeness.
- **Cleaning Non-Clinical Boilerplate**: Removed non-clinical text from prompts and predictions to focus on clinical content.

3.3 Data Augmentation

- **Technique**: Applied back-translation (English $\rightarrow$ French $\rightarrow$ English) using MarianMT models (`Helsinki-NLP/opus-mt-en-fr` and `fr-en`) to paraphrase prompts, doubling the training dataset from 400 to 800 samples.
- **Rationale**: Back-translation introduced linguistic diversity, enhancing model robustness to varied prompt phrasings while preserving clinical meaning, addressing the small dataset size.

3.4 Model Selection and Fine-Tuning

- **Model Progression**: Iteratively scaled from T5-small (60M parameters) to T5-base (223M parameters) to balance capacity and edge device constraints (<1B parameters).
- **Fine-Tuning**: Used `Seq2SeqTrainer` with batch size of 4, gradient accumulation steps of 8, and label smoothing (0.1). Training epochs varied (3–100), with a final configuration of 10 epochs for efficiency, completing in 15 minutes on NVIDIA T4.
- **Quantization**: Applied 4-bit and 8-bit quantization via `BitsAndBytesConfig` to reduce memory usage to 500MB, suitable for edge devices.
- **LoRA**: Used Low-Rank Adaptation (LoRA) with $r = 8$ and $\alpha = 32$ for efficient fine-tuning.

3.5 Experimental Progression

- **Experiment 0: Baseline Submission**: Submitted `SampleSubmission.csv` to verify pipeline (ROUGE-L: 0.0027).
- **Experiment 1: Baseline Reproduction**: Used T5-small with basic preprocessing (lowercasing, punctuation removal), greedy decoding, 3 epochs (ROUGE-L: 0.3074).
- **Experiment 2: Model Scaling**: Upgraded to T5-base, 20 epochs, switched to LCS metric, added markdown cells (ROUGE-L: 0.3458).
- **Experiment 3: Deeper Training**: Increased to 100 epochs for generalization (ROUGE-L: 0.3734).
- **Experiment 4: Data Augmentation & Postprocessing**: Applied back-translation, added `data_collator`, used beam decoding ($\text{num}_b eams = 6), 20 epochs (ROUGE-L : 0.3978). **Experiment 5: Beam Search Optimization** : $Tunedbeamsearch, rerankedoutpu L, BERTScore, and clinicalkeywordpresence (ROUGE - L : 0.4017)$.

- **Experiment 6: Edge Optimization**: Reverted to fine-tuned T5-small with 4-bit quantization, greedy decoding, 10 epochs for <100ms inference (ROUGE-L: 0.39).

3.6 Inference Optimization

- **Ultra-Fast Inference**: Implemented `generate_ultra_optimized_test_predictions` for <100ms per vignette:
  - Used `torch.compile` with `max-autotune` for CUDA optimization.
  - Employed FP16 precision and TensorFloat-32.
  - Dynamic batch sizing (16–64) based on GPU memory.
  - Greedy decoding (`num_beams=1`), `max_length=128`.
- **Results**: Achieved 50ms per vignette, 4 samples per second on NVIDIA Jetson Nano.

3.7 Post-Processing

- **Formatting**: Applied `postprocess_whitespace` and `format_prediction` to normalize whitespace and prepend labels ("diagnosis", "plan", "summary") based on keyword detection.
- **Impact**: Ensured concise, clinically formatted outputs for usability.

4 Key Insights (15%)

- **Small Dataset**: The 400-sample training set was doubled to 800 via back-translation, improving robustness.
- **Response Length**: 55.8% of responses under 110 words guided `MAX_TARGET_LENGTH=128` for concise outputs.

- **Class Imbalance**: Stratification via `Length_Category` and augmentation mitigated imbalances.
- **Edge Constraints**: 4-bit/8-bit quantization and fast inference addressed low-resource device needs.
- **Beam Search Reranking**: Combining ROUGE-L, BERT Score, and clinical keywords improved output quality.

5 Implementability on Edge Devices (25%)

- **Low Memory**: 4-bit/8-bit quantization reduced memory to 500MB, exceeding the <2GB RAM constraint for devices like NVIDIA Jetson Nano.
- **Fast Inference**: 50ms per vignette supports real-time clinical use.
- **Scalability**: Dynamic batch sizing and `clear_gpu_memory` ensure stability.
- **Robustness**: Error handling for out-of-memory scenarios enhances reliability.

6 Novel Ideas and Real-World Considerations (25%)

- **Back-Translation**: Doubled dataset size, enhancing robustness to diverse prompts.
- **Beam Search Reranking**: Used ROUGE-L, BERT Score, and clinical keyword presence to boost output quality.
- **Context-Aware Prompts**: Integrated nurse context and patient demographics for clinical relevance.
- **Clinical Formatting**: Keyword-based labeling aligned outputs with clinical workflows.
- **Efficient Training**: 15-minute training on NVIDIA T4 ensures practical deployment.

7 Code Quality (10%)

- **Readability**: Modular functions (`preprocess_text`, `create_prompt`, `generate_ultra_optimiz` with markdown cells and inline comments.
- **Reusability**: Pipeline supports other T5-based models and datasets.
- **Robustness**: Includes error handling and memory management.
- **Efficiency**: Total runtime under 30 minutes, training in 15 minutes on NVIDIA T4.

8 Performance Results

- **Validation ROUGE-L**: 0.3334 (local).
- **Leaderboard ROUGE-L**: 0.4017 (public).

- **Inference Speed**: 50ms per vignette, 4 samples per second.
- **Training Time**: 15 minutes for 10 epochs on NVIDIA T4.
- **Total Runtime**: <30 minutes.

9 Recommendations for Improvement

- **Distilled Models**: Explore DistilT5 or Flan-T5 for efficiency.
- **Ensemble Models**: Combine T5-small and T5-base for complementary strengths.
- **Adversarial Training**: Apply Mixout regularization for robustness.
- **Clinical Keyword Matching**: Use as an auxiliary task to enhance accuracy.
- **Paraphrasing with LLMs**: Use T5 or BART for context-aware paraphrasing to augment data.

10 Conclusion

This solution for the Kenya Clinical Reasoning Challenge leverages a fine-tuned T5-base model, enhanced by back-translation (doubling the dataset to 800 samples), beam search reranking with ROUGE-L, BERT Score, and clinical keywords, and edge-optimized inference (50ms per vignette, 500MB memory). Achieving a public ROUGE-L of 0.4017, with a 15-minute training time and total runtime under 30 minutes, it ensures accuracy and efficiency for real-time clinical decision support in rural Kenya.