# Release 2 Documentation

## Project Name: Cloud Diagram Translation & Reverse Engineering

## Team Members & Responsibilities:

* Shambhavi Singh - Refining structured prompt engineering to align with rules
* Akshita Arora -. Development of rule-based transformations for text-to-diagram generation.
* Aamish Samotra - Expansion of training dataset and evaluation of rule consistency.
* Lida Wei - Fine-tuning LLM to integrate structured rules for accurate diagram generation.

## 1. Objectives

The primary goal of Release 2 is to enhance text-to-diagram translation by implementing rule-based transformations for improved accuracy and consistency.

* Integrating structured rules to define cloud networking elements more precisely.
* Fine-tuning an LLM using the expanded dataset to ensure better adherence to networking logic.
* Improving Mermaid.js-based diagram generation by ensuring semantic correctness.
* Evaluating fine-tuned models against rule-based prompt engineering.

## 2. Deliverables

### 2.1 Core Functionalities

* Rule-Based Processing: Development of predefined transformation rules for converting text into structured network diagrams.
* Fine-Tuned Model: Training an LLM to incorporate rules and improve text-to-diagram accuracy.
* Enhanced Mermaid.js Output: Refining Mermaid.js diagram generation for better visual representation.
* Evaluation & Validation: Comparing rule-based transformations against structured prompt engineering.

## 3. Task Breakdown

### 3.1 Tasks Assigned to Team Members

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Description** | **Assigned To** | **Expected Outcome** |
| Rule Definition | Develop a structured rule set for diagram generation. | Team Member 2 | Documented rule-based transformations. |
| Prompt Refinement | Modify structured prompts to align with rules. | Team Member 1 | Improved text prompts. |
| Dataset Expansion | Extend dataset with rule-based networking scenarios. | Team Member 3 | Enriched dataset for training. |
| LLM Fine-Tuning | Train and evaluate model with rule-based inputs. | Team Member 4 | Enhanced model performance. |
| Mermaid.js Rendering | Validate diagram output against rules. | Team Member 1 & 2 | Optimized visual representation. |
| Evaluation Metrics | Define and implement accuracy checks. | Team Member 3 & 4 | Performance comparison report. |

## 4. Testing & Validation

* Rule Compliance Check: Ensuring LLM-generated diagrams follow the predefined rules.
* Dataset Quality Assessment: Evaluating the diversity and correctness of networking scenarios.
* Diagram Accuracy: Measuring correctness in generated cloud networking diagrams.
* Fine-Tuning Effectiveness: Comparing performance before and after model fine-tuning.

## 5. Risks & Mitigation Strategies

|  |  |  |
| --- | --- | --- |
| **Risk** | **Impact** | **Mitigation Strategy** |
| Incorrect rule application | High | Implement validation checks for rule compliance. |
| Training inconsistencies | Medium | Regular fine-tuning evaluation with test cases. |
| Mermaid.js limitations | Medium | Enhance rule set for more structured diagram outputs. |
| Model generalization issues | High | Expand dataset with diverse network configurations. |

## 6. Next Steps

* Optimize Rule-Based Processing: Refine transformation logic to improve text-to-diagram accuracy.
* Expand Training Data: Incorporate additional complex networking scenarios.
* Automate Evaluation Metrics: Implement automated quality checks for generated diagrams.
* Client Feedback Integration: Adjust rule-based transformations based on real-world use cases.

## 7. Conclusion

Release 2 significantly enhances text-to-diagram translation by introducing rule-based transformations and fine-tuned model training. By incorporating structured networking logic, refining Mermaid.js outputs, and optimizing prompt engineering, the team has created a more accurate and scalable cloud diagram generation process. The next phase will focus on refining rule automation, expanding dataset diversity, and further optimizing model accuracy.