An Adaptive Neuro-Symbolic Architecture for Explainable Relation Extraction

Brief Milestone 2 Report

November 2025

1 Introduction and Background

Relation Extraction (RE) is a fundamental task in Natural Language Processing, serving as the backbone for building Knowledge Graphs, answering questions, and summarizing text. The goal is to identify semantic relationships between entities in a sentence (e.g., distinguishing that *"The engine is **in** the car"* represents a `Component-Whole` relationship).

While modern Deep Learning (DL) models have achieved state-of-the-art performance on these tasks, their application in high-stakes domains—such as biomedical research, legal contract analysis, or engineering—is hindered by a critical limitation: the "Black Box" problem.

2 Problem Statement (The Gap)

Current approaches to Relation Extraction face a binary trade-off between **Accuracy** and **Explainability**:

- **The Neural Approach (e.g., BERT, RoBERTa):**
  - *Pros:* High recall; understands semantic context and synonyms; robust to noise.
  - *Cons:* Completely opaque. It provides a probability score but no linguistic justification for *why* a relationship was predicted. It cannot be easily audited or debugged.
- **The Symbolic Approach (e.g., Rule-based systems, spaCy):**
  - *Pros:* High precision; fully transparent; intrinsically explainable (e.g., "Predicted X because of dependency path Y").
  - *Cons:* Brittle. It suffers from low recall because it cannot handle variations in phrasing not explicitly hard-coded by humans.

The Gap: There is currently a lack of systems that possess the **generalization capabilities** of neural networks while maintaining the **intrinsic explainability** and **auditability** of symbolic rules.

3 Proposed Solution

This project proposes an **Adaptive Neuro-Symbolic Architecture**. Instead of treating Rules and Neural Networks as competing methods, we integrate them into a cooperative loop.

The core innovation is **Model-Guided Rule Induction**. The system uses a Rule Database as the primary, trusted classifier. When rules fail, it uses a BERT model not just to make a prediction, but to **teach the system a new rule**, converting neural intuition into explicit symbolic logic.

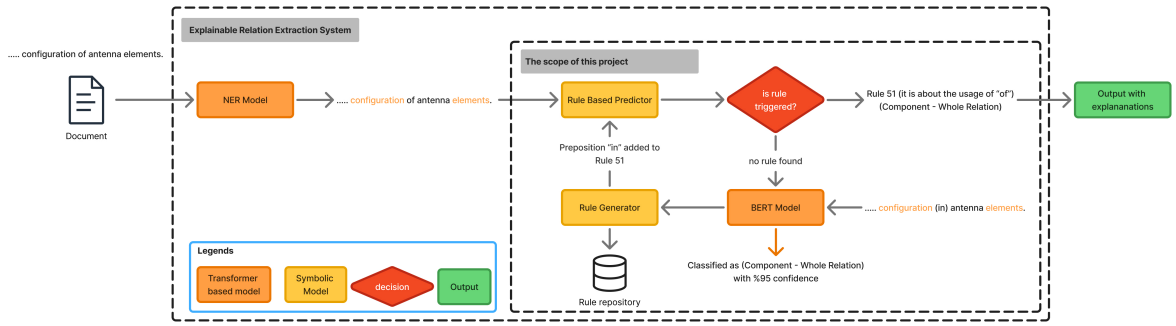

Figure 1: The proposed neuro-symbolic feedback loop.

4 Methodology

The project will be executed in three distinct phases.

4.1 Phase 1: Foundation (Cold Start)

Objective: Establish a high-precision symbolic baseline and a high-recall neural safety net.

- **Symbolic Initialization:** We will generate rules directly from the annotated entities within the training set. By analyzing the dependency paths connecting these entities, we will identify frequent syntactic archetypes (e.g., `PREP_PHRASE`) and auto-generate an initial set of spaCy dependency rules.
- **Neural Training:** Parallel to this, a BERT-based classifier will be fine-tuned on the labeled training data to learn the semantic context of the relations.

4.2 Phase 2: Adaptive Inference & Rule Induction

Objective: Handle unseen data and dynamically expand the rule set. This phase implements the novel "Rescue & Learn" logic on dev/test data:

1. **Step A: The Symbolic Filter (Priority Check)**
   - The system first attempts to classify the instance using the Rule Database.
   - *Result:* If a match is found, the prediction is accepted with **100% explainability** (e.g., *"Classified as Component-Whole due to rule PREP_OF matching dependency path* $X \rightarrow of \rightarrow Y$*"*).
2. **Step B: The Neural Rescue (Gap Analysis)**
   - If no rule matches, the system queries the BERT model.
   - If BERT predicts "Other" or has low confidence, the instance is discarded.
   - If BERT predicts a specific relation (e.g., `Component-Whole`) with **high confidence**, the system flags this as a **"Knowledge Gap."**
3. **Step C: Just-in-Time Induction**
   - The system analyzes the dependency path of this "missed" example.
   - It determines if the path fits a known archetype but uses unknown vocabulary (e.g., a new preposition like "in").
   - *Action:* The system dynamically updates the Rule Database so that future instances are caught by the symbolic layer.

4.3 Phase 3: Evaluation and Iteration

Objective: Assess performance and explainability coverage.

- The system will be evaluated on the SemEval test set.
- **Metrics:**
  - Standard: Precision, Recall, F1-Score.
  - **Novel Metric: Explainability Coverage** (The percentage of true positive predictions that are backed by a symbolic rule vs. those remaining as "neural guesses").
  - Rule Quality: Human verification of the auto-induced rules to ensure they represent valid linguistic patterns.

5 Expected Impact and Deliverables

1. **An Explainable RE System:** A pipeline where the majority of decisions are traceable to linguistic syntax.
2. **Dynamic Knowledge Base:** A set of dependency rules that grows and adapts to the data distribution automatically, reducing the need for manual rule writing.
3. **Auditability:** A framework where errors can be fixed by removing a specific rule, rather than vaguely retraining a black box.

This project moves beyond "trusting the AI" to "verifying the AI," making high-performance NLP viable for domains requiring rigorous transparency.