Project: Explainable Relation Extraction - Token13
Management Summary - 25.01.2026

1. Project Overview

Identifying relationships between entities (e.g., Company A acquired Company B) is essential in text-processing applications, a task known as Relation Extraction (RE). Although deep learning models such as BERT perform well, they often lack transparency. This opacity creates challenges in enterprise settings, where clear decision justifications are required for compliance and debugging.

Objective: We aimed to develop a rule-based relation extraction system that combines human-interpretable logic with automated, data-driven methods. Rather than manually creating extensive rule sets, we designed an engine that automatically mines labeled training data to identify linguistic patterns and convert them into rules. This approach produces an explainable, deterministic, and highly configurable system that provides a transparent alternative to neural network-based models.

2. Challenges

Developing an automated rule-based system involves specific linguistic and operational challenges:

- **Linguistic ambiguity:** Natural language is rarely precise. A preposition like "of" can signal ownership ("wheel of the car"), origin ("wine of France"), or composition ("statue of gold"). Teaching a rule-based system to distinguish these nuances without semantic "common sense" is difficult.
- **The Precision-Recall Trade-off:** We faced a strategic choice between being "often right but missing some data" (High Precision) or "capturing everything but making many errors" (High Recall). We choose precision to ensure trust in the output.
- **Data sparsity:** Some relation types occur frequently, while others are rare. Pattern mining becomes unreliable for rare relation types because there is insufficient evidence to form robust rules.

3. External Resources

We used the SemEval 2010 Task 8 dataset for training and evaluation of annotated sentence-level relations. Our rule-based system leverages SpaCy’s Matchers for pattern detection, enhanced by semantic knowledge from WordNet (synsets/hypernym-based type hints) and FrameNet (frame-level predicate interpretations). Additionally, we fine-tuned RoBERTa as a neural baseline on the same dataset to provide a performance reference for the explainable rule-based approach.

4. The Solution

We implemented an Automated Rule Discovery Engine. Rather than relying on human linguists to manually code rules, our solution works as follows:

- **Pattern Mining:** The system scans the training text and extracts features connecting two entities. It looks for Lexical Patterns (specific words like "caused

by", "inside of") and Grammatical Patterns (how words are connected in the sentence structure).
- **Quality Filtering:** Every discovered pattern is scored based on its statistical reliability. We established a strict quality gate: a pattern becomes a "Rule" only if it is correct at least 60% of the time it appears.
- **Deterministic Decision Making:** When processing new text, the system applies these rules in a ranked order. The most precise rule "wins." If no strong rule matches, the system conservatively categorizes the relationship as "Other."

Key Advantages: can be given as follows;

- **Transparency:** Every prediction comes with an explanation (e.g., "Classified as Cause-Effect because the phrase 'triggered by' links the entities").
- **Speed:** The system is computationally lightweight compared to neural networks, requiring 2–3 minutes for training and 1–2 minutes for prediction, making it signifi
- **Customizability:** Stakeholders can easily add, remove, or modify specific rules without retraining the entire system.

The system improved from **0.50 Accuracy**, **0.59 Precision, 0.44 Macro-F1** at Milestone 2, to **0.60 Accuracy**, **0.66 Precision, 0.56 Macro-F1** at the end of Milestone 3.

5. Limitations and Discussion

While the system succeeds in being transparent and high-precision, it has limitations inherent to rule-based approaches:

Conservative Nature (Low Recall): The system is designed to be "safe." If a sentence uses phrasing the system hasn't seen before, it will likely default to "Other" rather than guessing. This means we miss valid relationships (False Negatives) to avoid making wrong calls.

Difficulty with Abstract Relations: The system excels at concrete relationships marked by specific words (e.g., Entity-Destination marked by "into" or "towards"). It struggles with abstract concepts like Member-Collection (e.g., "one of the crew members"), where the relationship is implied by meaning rather than specific trigger words.

Performance Metrics: On the test set, the system achieved an accuracy of roughly 0.60. While lower than state-of-the-art neural models (which reach ~0.80), our errors are predictable and explainable, whereas neural errors are often harder to interpret and diagnose.