Task 6: Researching other Logics + LLMs

Task 6 – Understanding Datalog

Haardhik Mudagere Anil

Date: 19 May 2025

What is Datalog?

- Datalog is a declarative logic programming language.
- Subset of Prolog: designed for database queries and reasoning.
- Uses facts, rules, and queries – no function symbols allowed.
- Well-suited for finite, recursive queries over relational data.

Key Characteristics of Datalog

- Bottom-up evaluation (vs. Prolog's top-down).
- Guaranteed to terminate for finite programs.
- No complex terms: only constants and variables.
- Supports recursion but not negation by default.

Datalog Syntax Overview

- Facts: parent(john, mary).
- Rules: ancestor(X, Y) :- parent(X, Y).
- Queries: ?- ancestor(john, mary).
- All variables are universally quantified in rules.

Datalog vs Prolog

- No backtracking in Datalog; uses fixed-point computation.
- No function symbols or complex terms.
- Prolog allows cuts and control mechanisms; Datalog does not.
- Datalog is non-Turing complete (more predictable).

Applications of Datalog

- Static program analysis (e.g., Soufflé).
- Database query optimization.
- Data provenance and auditing.
- Knowledge graph reasoning.

Tools and Implementations

- Soufflé: High-performance Datalog engine for program analysis.
- PyDatalog: Python-embedded Datalog engine.
- LogicBlox: Commercial Datalog-based system.
- Datomic: Influenced by Datalog, used in distributed databases.

Why Datalog for LLMs?

- Symbolic reasoning over structured relational data.
- Well-defined semantics, easy to ground into vector spaces.
- Used in symbolic submodules for interpretability.
- Integration with neural models via neuro-symbolic methods.