# Probabilistic Relaxed Unification Formalism and its Application in Question Answering

PhD Thesis Defence

August 28, 2008

Tony Abou-Assaleh

Faculty of Computer Science
Dalhousie University

# Agenda

- Introduction
- Related Work
- Problem Description
- Probabilistic Relaxed Unification
- RUQA System Prototype
- Empirical Validation
- Conclusion

# Agenda

- **Introduction**
- Related Work
- Problem Description
- Probabilistic Relaxed Unification
- RUQA System Prototype
- Empirical Validation
- Conclusion

# Introduction

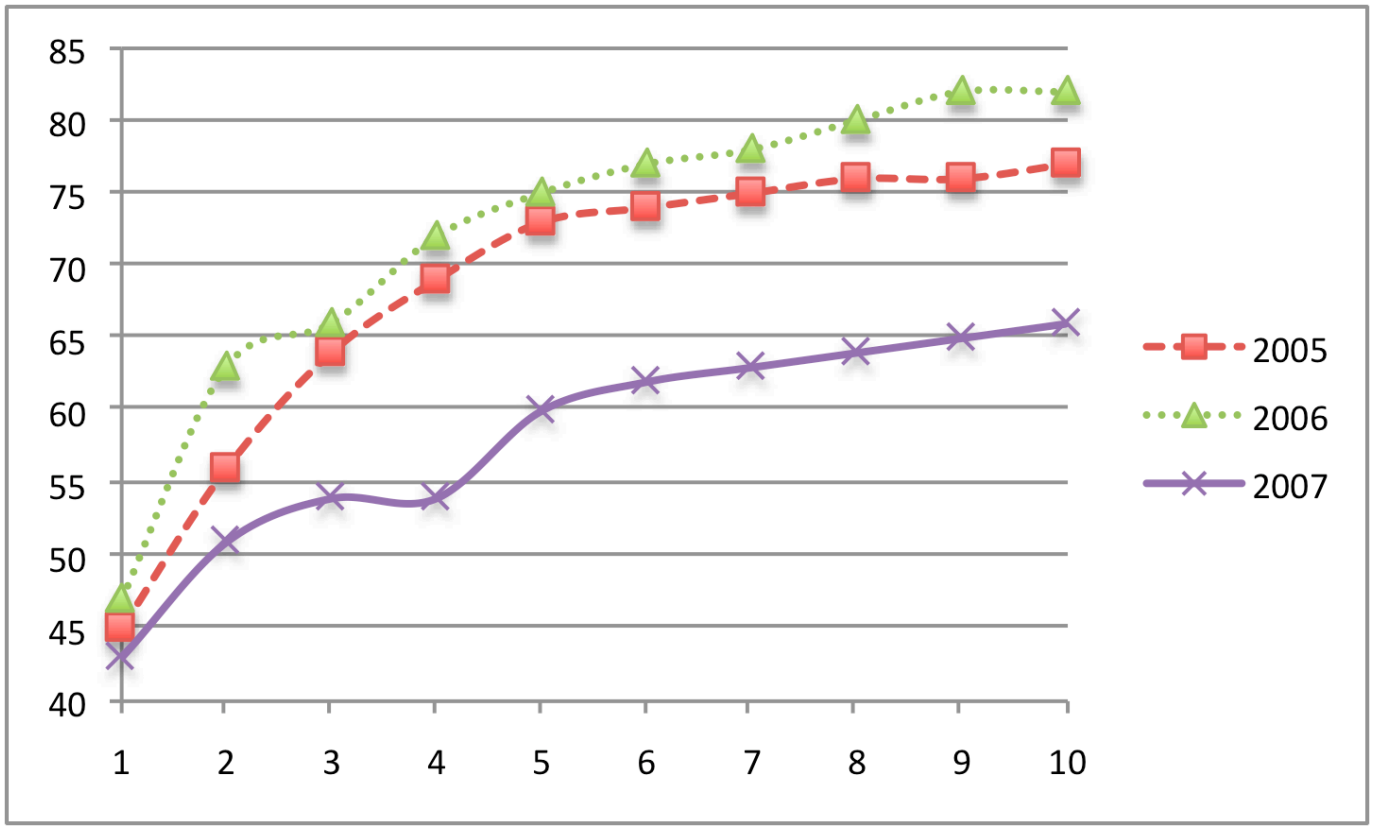
- What is unification?
  - Binary function: success / failure
  - Two-way matching
  - Make the two operands the same
- Applications:
  - Logic Programming
  - Theorem Proving
  - AI: Expert Systems, Game Playing
  - NLP: Unification Grammars

# Motivation

- Natural Language Processing
  - Statistical vs. Unification based
- Classical Unification Fails
  - Imperfect real-world data
- HPSG is imperfect
  - Grammar, lexicon
- Question Answering
  - Can we do better?

# Jellyfish QA System

Correctly answered questions vs. number of candidate answers



# Thesis Statement

- Useful information can be inferred from unifying mismatched terms by quantifying the degree of the mismatch

# Objectives

- Relaxed Unification Formalism and Algorithm
- Evaluation Function
- Question-Answering System Prototype
- Empirical Validation

# Agenda

- Introduction
- **Related Work**
- Problem Description
- Probabilistic Relaxed Unification
- RUQA System Prototype
- Empirical Validation
- Conclusion

# Related Work

- Classical Unification
  - Robinson 1965, Knight 1989
- Uncertainty in Logic
  - Lee 1972, Bacchus 1990
- Logic Programming
  - Cox 1977
- Multi-valued Attributes
  - Pollard & Moshier 1990, Carpenter 1992

# Related Work

- Relaxed Unification
  - Abou-Assaleh 2002
- Head-driven Phrase Structure Grammar (HPSG)
  - Pollard & Sag 1994
- Question Answering (QA)
  - TREC 1999 – 2007
  - TAC 2008
  - Lymba 2007 (PowerAnswer 4)

# Agenda

- Introduction
- Related Work
- **Problem Description**
- Probabilistic Relaxed Unification
- RUQA System Prototype
- Empirical Validation
- Conclusion

# Problem Description

- Unification of Conflicting Graphs
  - Graph Unification
    - Framework, formalism, and algorithm
  - Evaluation Function
    - Structure and data dependent
    - Granular
    - Computable

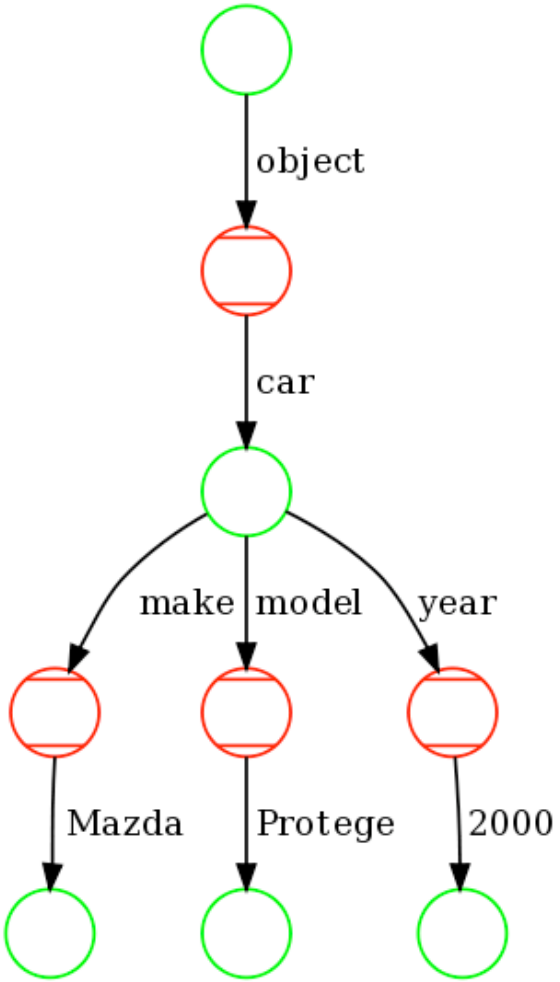
# Problem Description

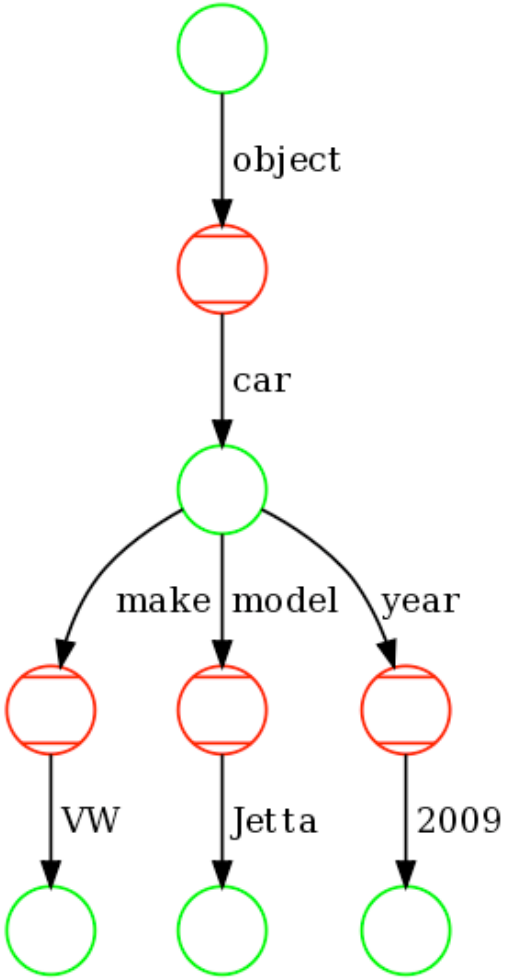
- Candidate Answer Ranking in Factoid QA
  - Question + candidate answers
  - Semantic Representation
  - Relaxed unification
  - Rank correct answer highest

# Agenda

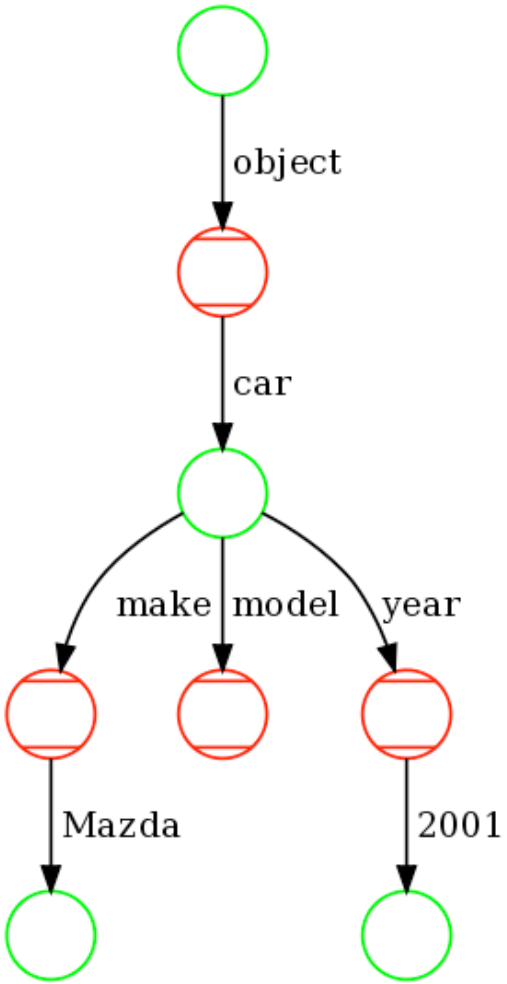
- Introduction
- Related Work
- Problem Description
- **Probabilistic Relaxed Unification**
- RUQA System Prototype
- Empirical Validation
- Conclusion

# I have 2 cars: Protégé and Jetta

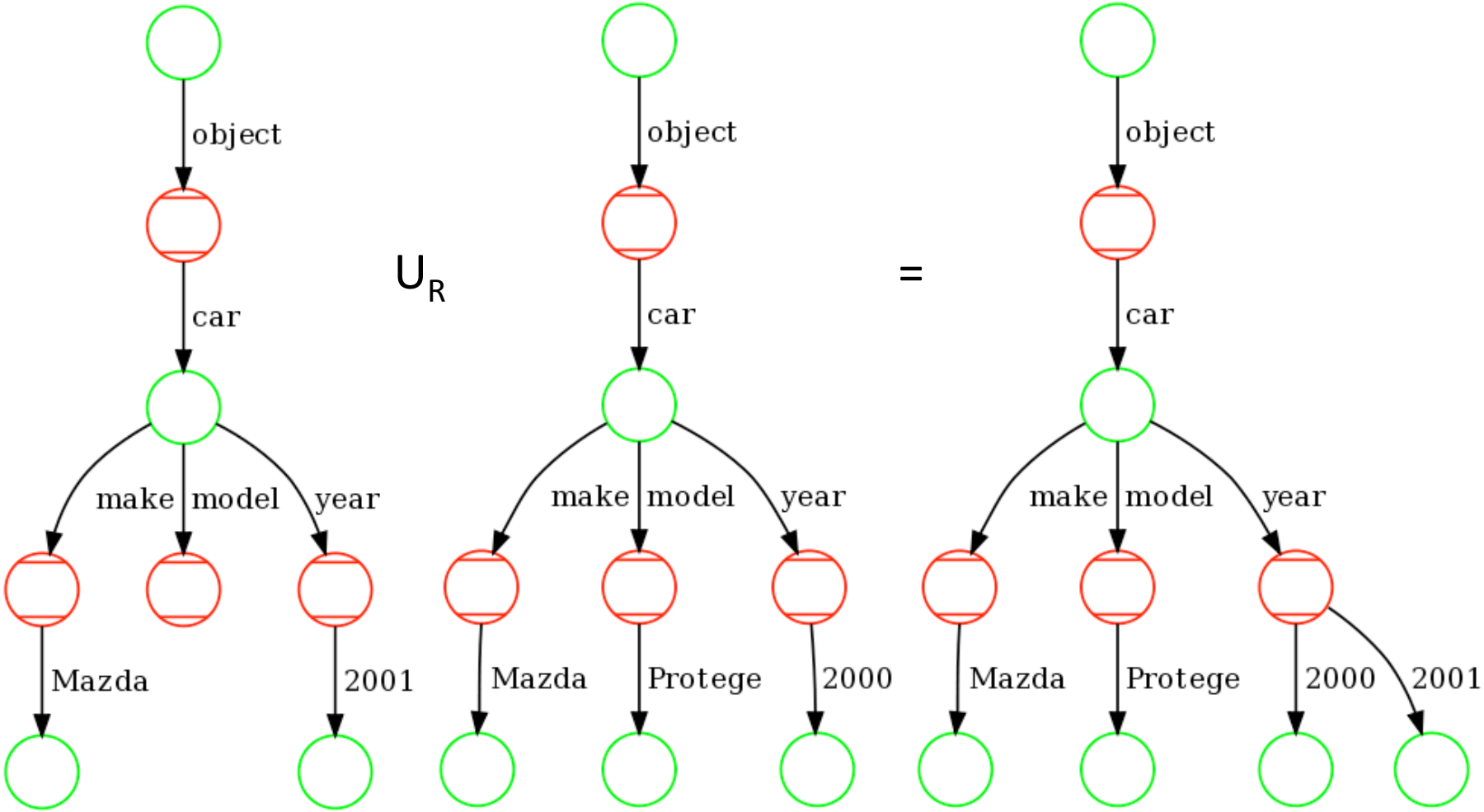


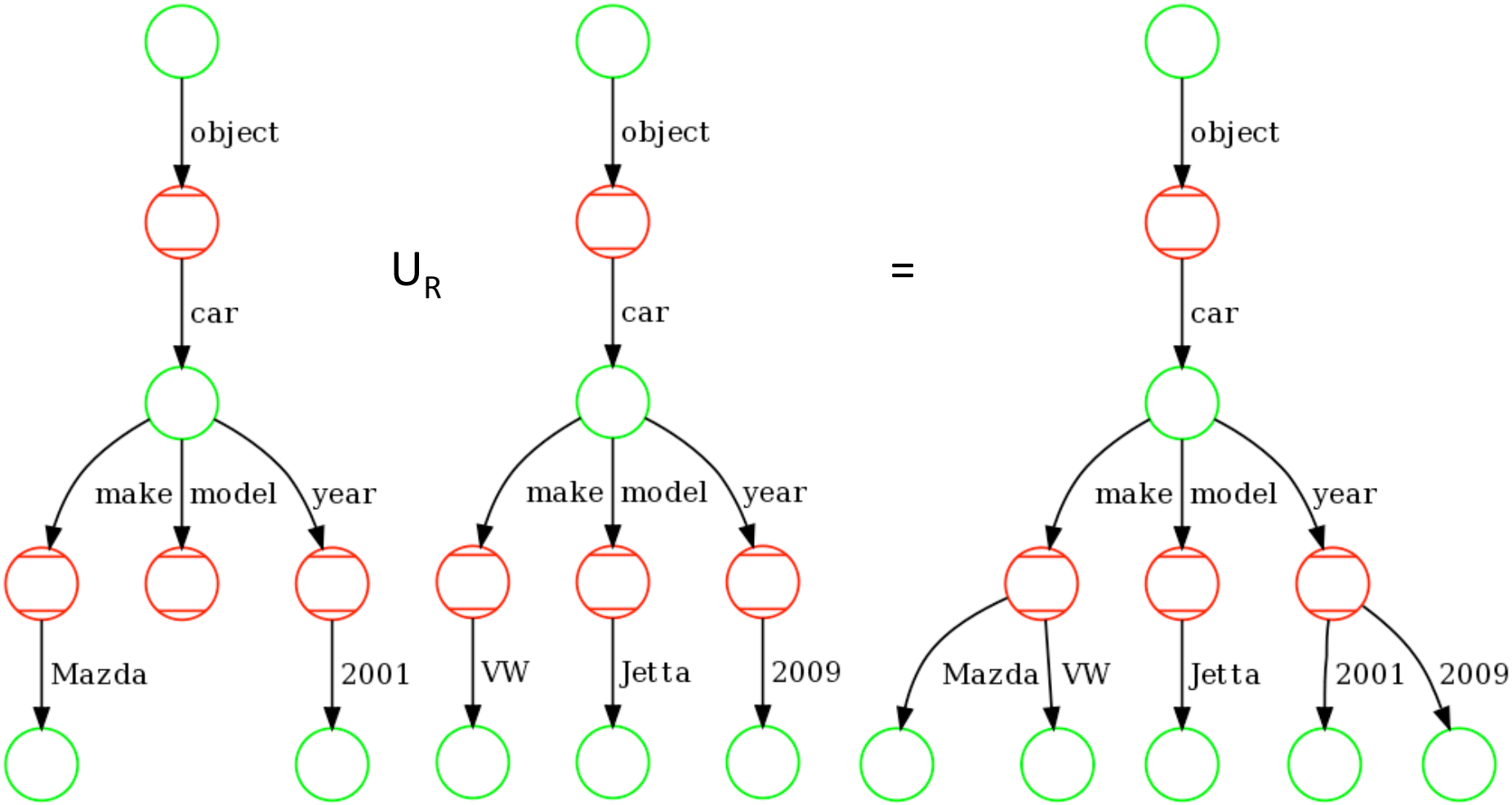
# Change policy on Mazda 2001



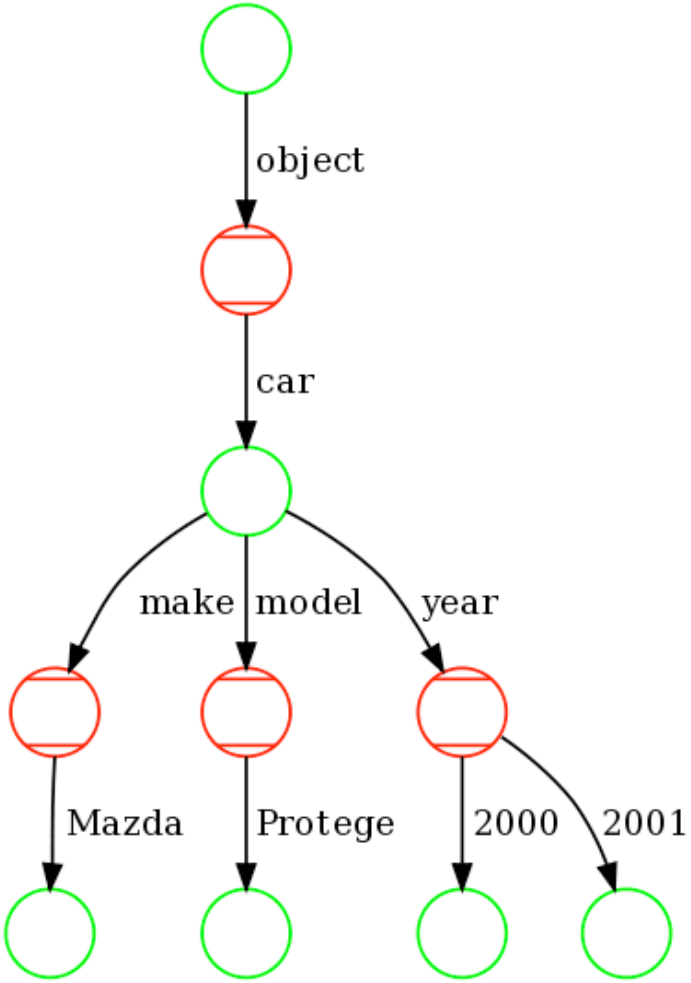
# Mazda 2001 =? Mazda Protégé 2000

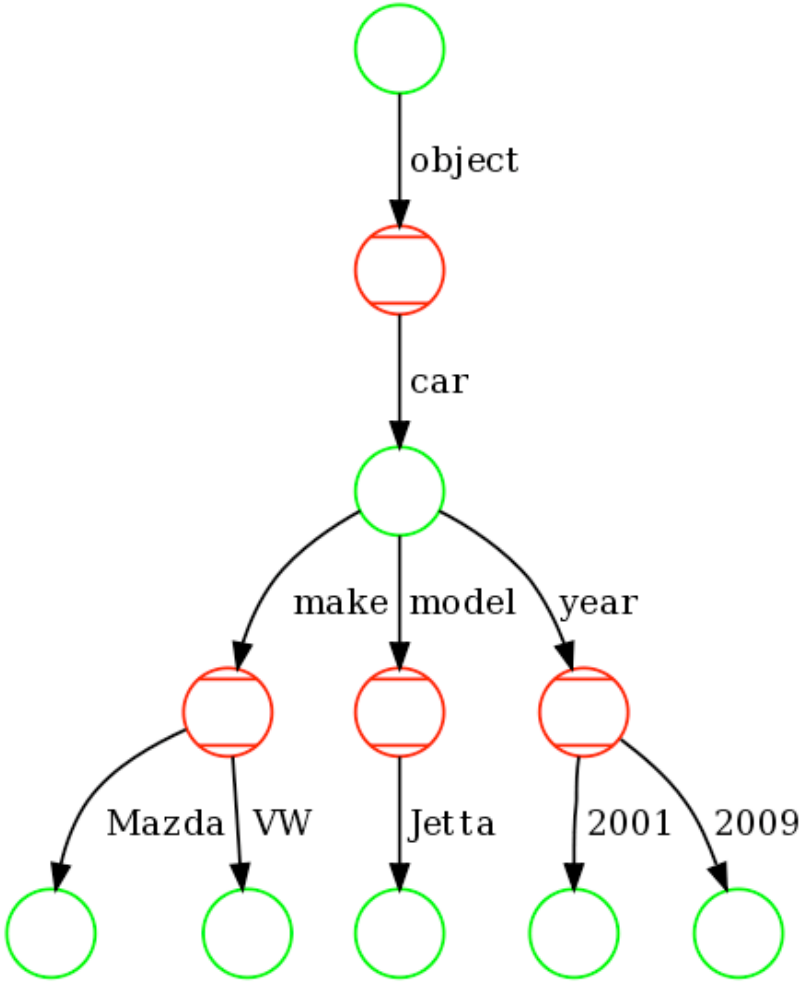


# Mazda 2001 =? VW Jetta 2009



# Protégé vs. Jetta

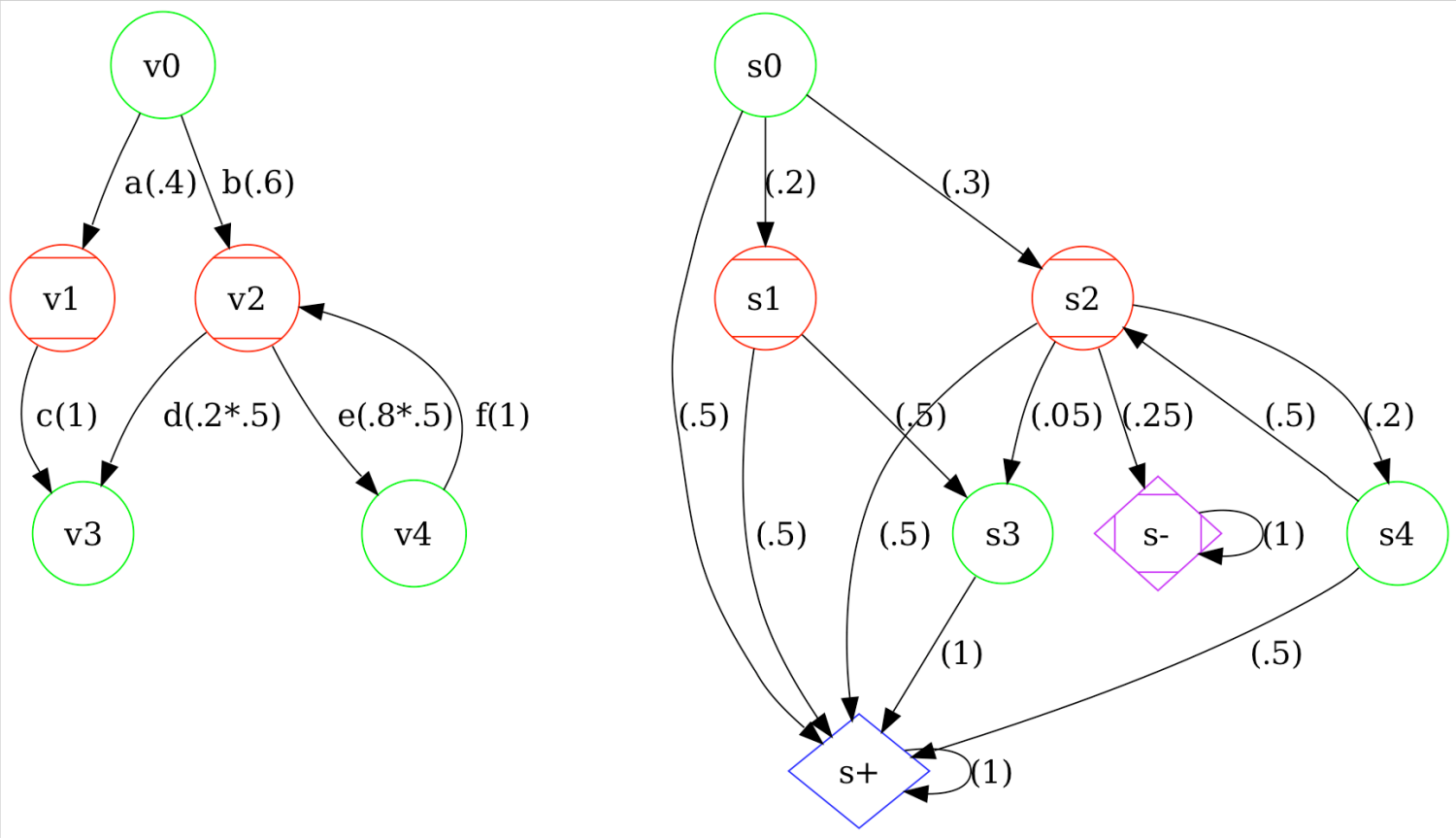


# Probabilistic Relaxed Terms

- $T = < V, r, E, \Gamma, \alpha, \beta, \gamma, \theta, \omega >$
  - V: Vertices, r: root
  - E: Edges
  - Γ: function symbols
  - α: E → V (source node)
  - β: E → V (end node)
  - γ: E → Γ (edge labels)
  - θ: V → {attribute, value}
  - ω: E → [0,1] (edge weight)

# Relaxed Terms as Finite Markov Chains



# Correctness Function

- Probability Transition Matrix *P*

$$P = \begin{pmatrix} I & O \\ R & Q \end{pmatrix}$$

- Steady state probability matrix *B* for absorbing states

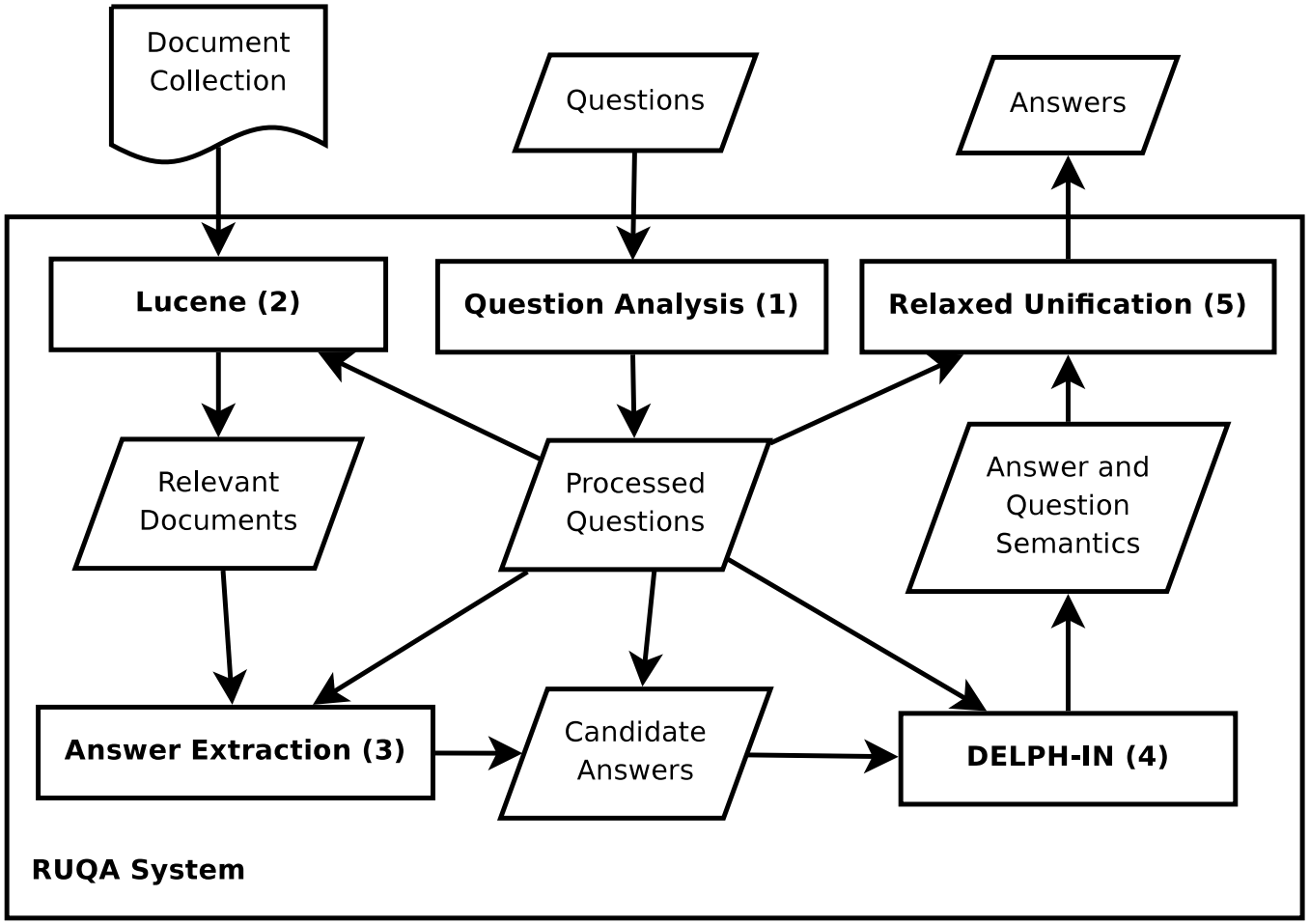
$$B = (I - Q)^{-1} R$$

- Closed form computation

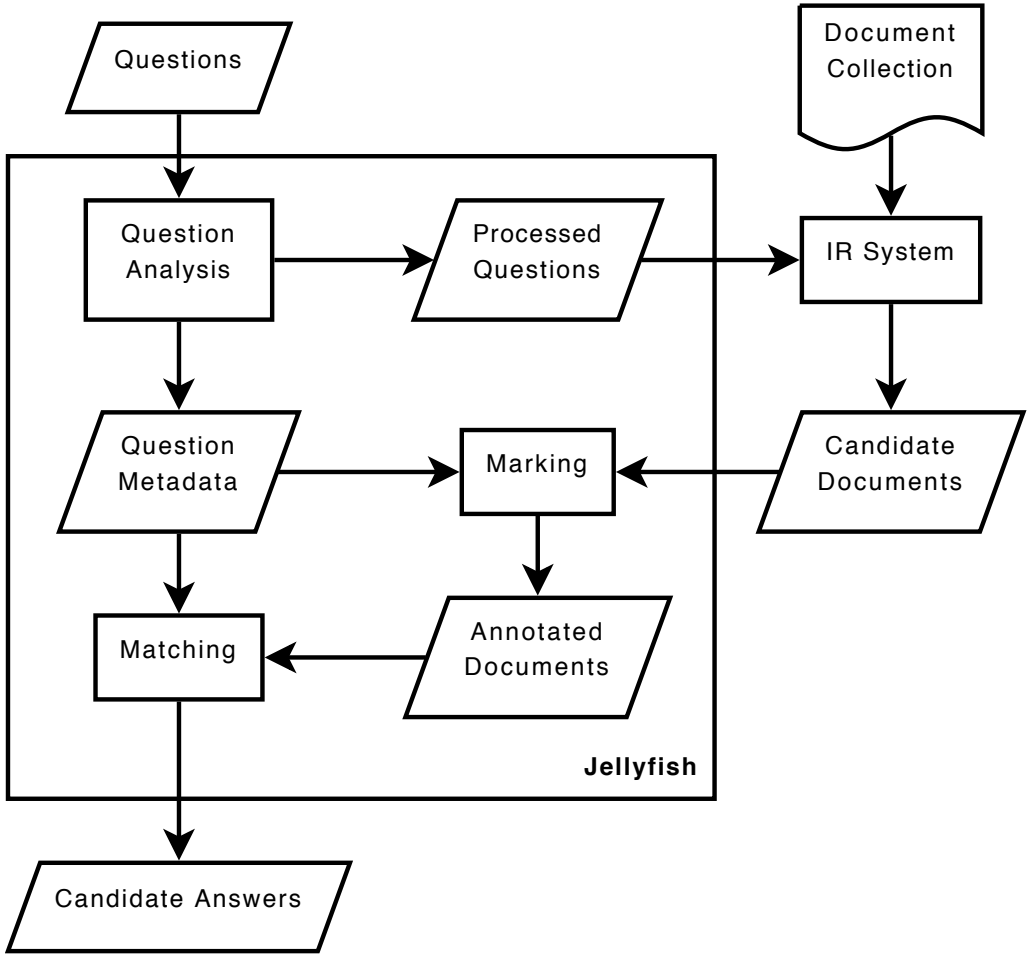
# Agenda

- Introduction
- Related Work
- Problem Description
- Probabilistic Relaxed Unification
- **RUQA System Prototype**
- Empirical Validation
- Conclusion

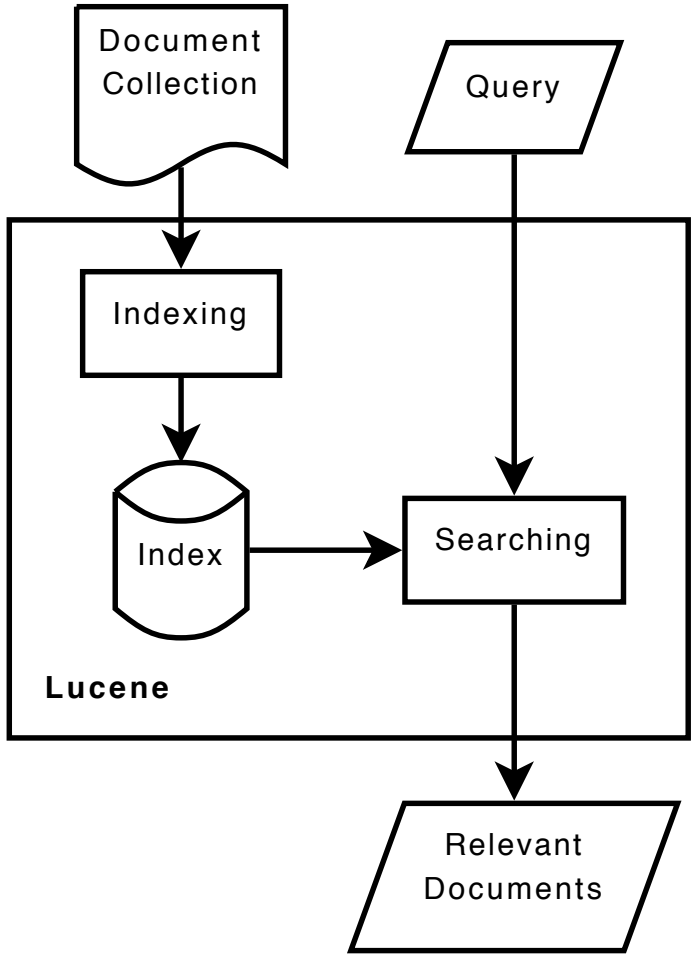
# RUQA Architecture



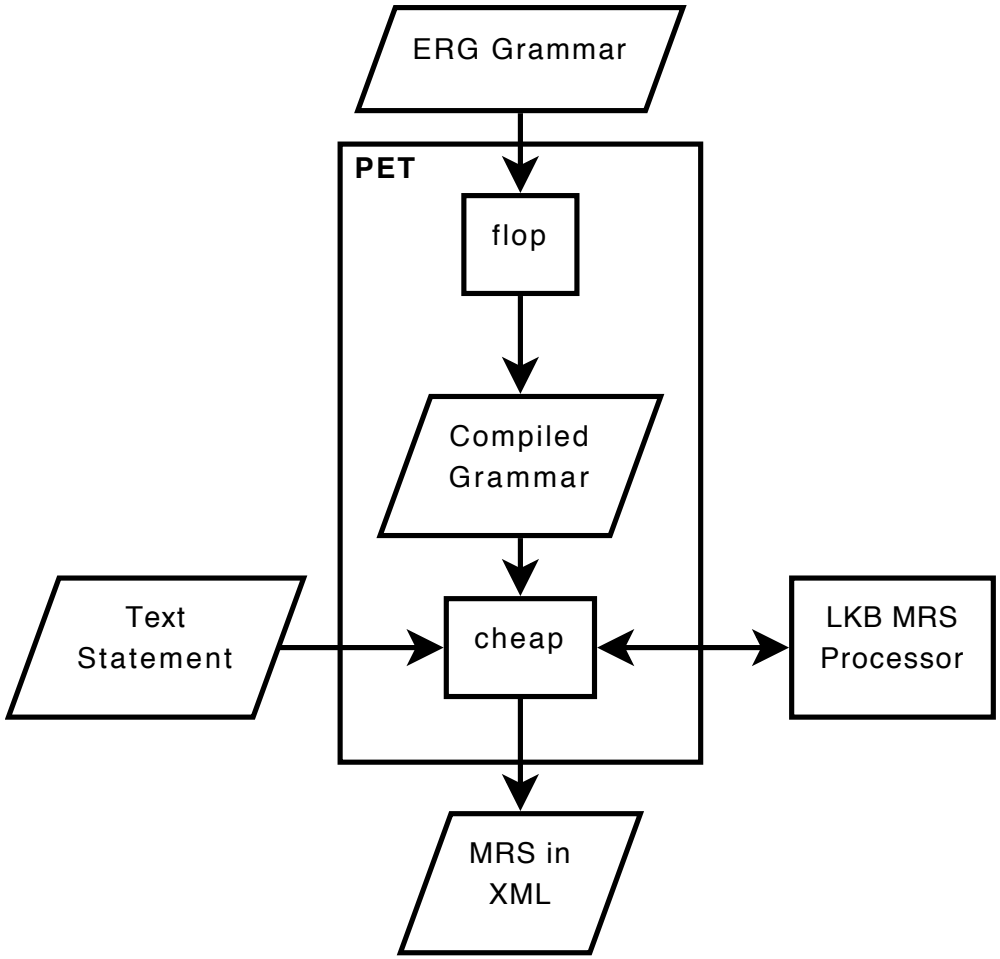
# Jellyfish



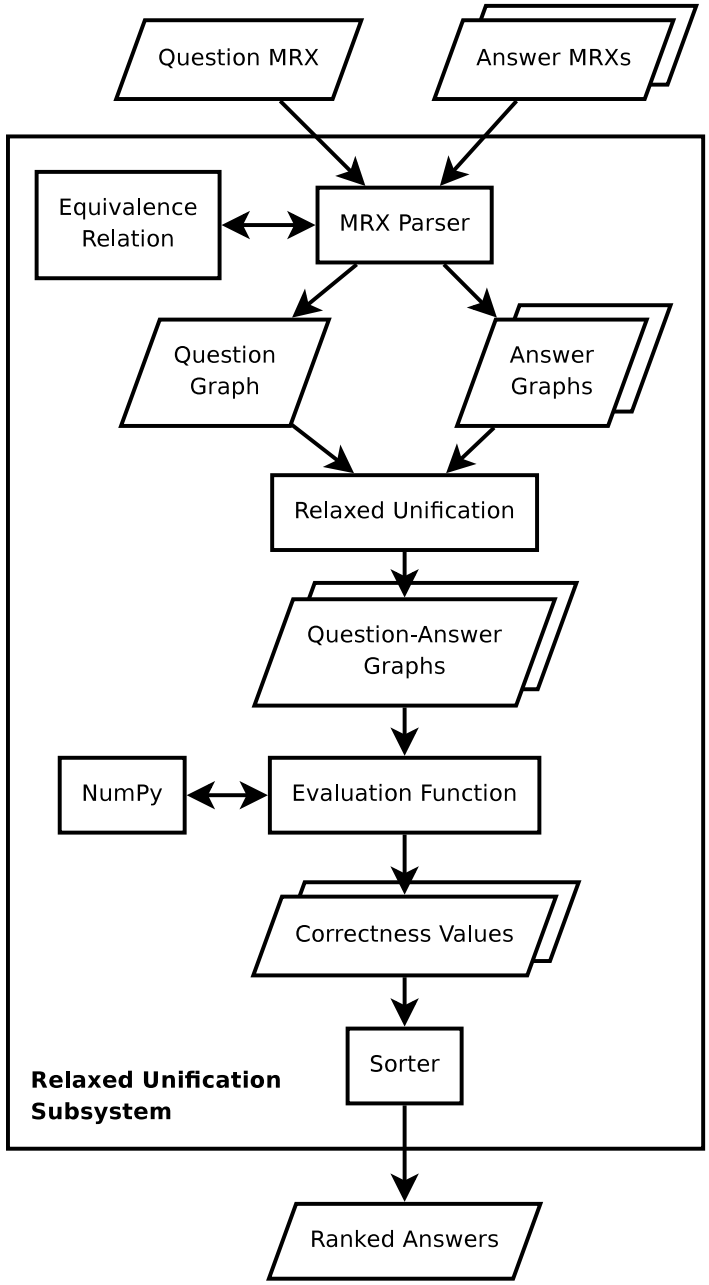
# Apache Lucene



# DELPH-IN Tools and Resources

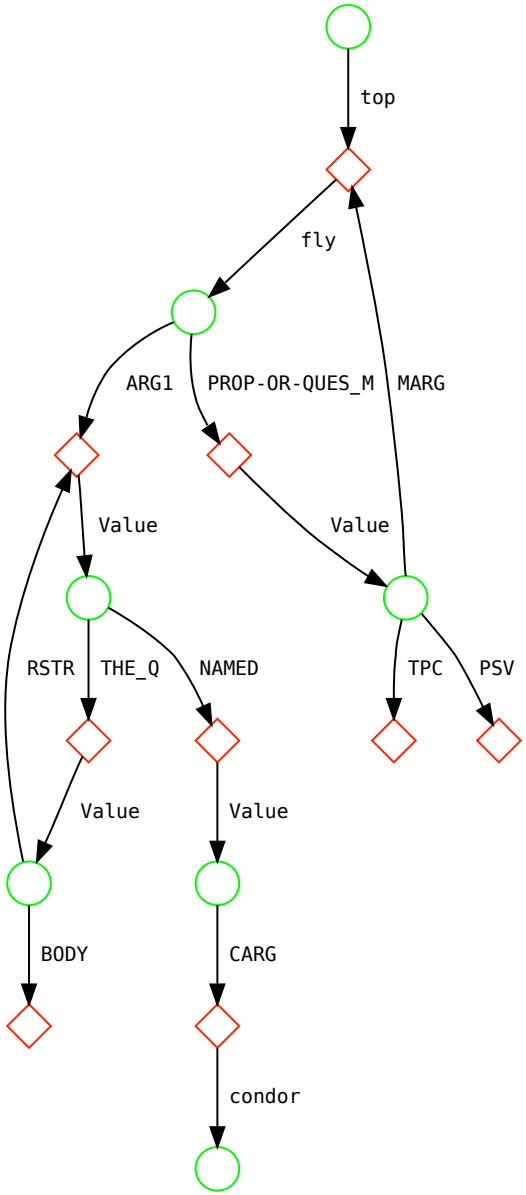


# Relaxed Unification Module



# The Condor Flies

[ LTOP: h1
  INDEX: e2 [ e TENSE: PRES MOOD: INDICATIVE PROG: - PERF: - ]
  RELS: <
      [ **prpstn_m_rel**
        LBL: h1
        ARG0: e2
        MARG: h3
        PSV: u4
        TPC: u5 ]
      [ **_the_q_rel**
        LBL: h6
        ARG0: x9 [ x PERS: 3 NUM: SG DIV: - ]
        RSTR: h8
        BODY: h7 ]
      [ **named_rel**
        LBL: h10
        ARG0: x9
        CARG: "condor" ]
      [ **"_fly_v_1_rel"**
        LBL: h11
        ARG0: e2
        ARG1: x9 ] >
  HCONS: < h3 qeq h11 h8 qeq h10 > ]



# Agenda

- Introduction
- Related Work
- Problem Description
- Probabilistic Relaxed Unification
- RUQA System Prototype
- **Empirical Validation**
- Conclusion

# Empirical Validation

- Empirical validation of a computer program (software) is a substantiation that the computer program provides satisfactory output on real-world data within its intended domain.

# Validation Procedure

- Data: TREC Questions + AQUAINT articles
- Baseline system: Jellyfish
- Validation cases selection
- Semantic Parsing
- Relaxed Unification

# Validation Case

- Question:
  - In what year did Catherine the Great begin buying for the museum?
- Answers:
  1. who took over in 1992
  2. Catherine began buying for her museum in 1764
  3. its major expansion gets under way in 2001
  4. the defense that turned back Nazi armies at the gates of the city in 1941
  5. Catherine Palace in Tasarkoe Selo, almost completely destroyed in 1944.

# Final Ranks

| Final Rank | Answer | Correctness | Original Rank |
|---|---|---|---|
| **1** | **1764** | **0.999993797786** | **2** |
| 2 | 2001 | 0.874760951746 | 3 |
| 3 | 1941 | 0.874760951746 | 4 |
| 4 | 1944 | 0.874689616504 | 5 |
| 5 | 1992 | 0.874516224330 | 1 |

# Validation Summary

1. Detailed walkthrough
2. Granularity of correctness function
3. Word similarity measure
4. Perfect match
5. Limitation of current system

# Agenda

- Introduction
- Related Work
- Problem Description
- Probabilistic Relaxed Unification
- RUQA System Prototype
- Empirical Validation
- **Conclusion**

# Contributions

- Relaxed Unification Formalism
  - Definitions, algorithm, types of evaluation functions
- Probabilistic Relaxed Unification
  - Weighted features, probabilistic correctness function
- Relaxed Unification Question Answering
  - Modular framework, advanced prototype
- Empirical Validation
  - Real data

# Open Questions and Future Work

- Better time complexity for relaxed unification
- Evaluate correctness during unification
- Non-destructive relaxed unification
- Relaxed Logic
- Relaxed HPSG
- Fully automated RUQA system
- Graph similarity

# Thank you!

Question or comments?