Question Answering and AnswerFinder

Diego Mollá

Centre for Language Technology
Department of Computer Science
Macquarie University

Athens, 4 July 2007

Programme

Programme

Who Are We? - Centre for Language Technology

Centre for Language Technology

- http://www.clt.mq.edu.au
- Located at Macquarie University, Sydney
- Members:
  - Director: Robert Dale
  - Core academic staff: Steve Cassidy, Mark Dras, Diego Mollá, Debbie Richards, Rolf Schwitter, Paul Watters
  - Honorary associate members: Dominique Estival, Mark Lauer, Cécile Paris
  - 4 Research Assistants
  - 11 PhD Students

Who Are We? - AnswerFinder

- http://www.ics.mq.edu.au/~diego/answerfinder/
- Funded by the Australian Research Council (ARC Discovery Grant)
- Members:
  1. Diego Mollá (Project Director, Chief Investigator)
     - Design, Sentence Representation, QA Rules
  2. Robert Dale (Chief Investigator)
     - Generation, Summarisation
  3. Menno van Zaanen (Research Associate)
     - Implementation, Machine Learning
  4. Luiz Augusto Pizzato (PhD student)
     - Document Retrieval, Web
  5. Daniel Smith (Research Programmer)
     - Named Entity Recognition

Programme

What is Text-based Question Answering (QA)?

Steps in Text-based Question Answering

1. Accepts any English question
2. Searches a collection of text documents
3. Returns an answer to the user question

What Questions can we Answer?

- Factoid
  - Text
  - Cross-lingual
  - Speech transcripts
- Query-driven summaries

Issues in Question Answering

- Size of the corpus
- Types of questions
- Types of expected answers
- Paraphrases of questions and answers
- Locating the exact answer

Issues in Question Answering

- Size of the corpus
- Types of questions
- Types of expected answers
- Paraphrases of questions and answers
- Locating the exact answer

Issues in Question Answering

- Size of the corpus
- Types of questions
- Types of expected answers
- Paraphrases of questions and answers
- Locating the exact answer

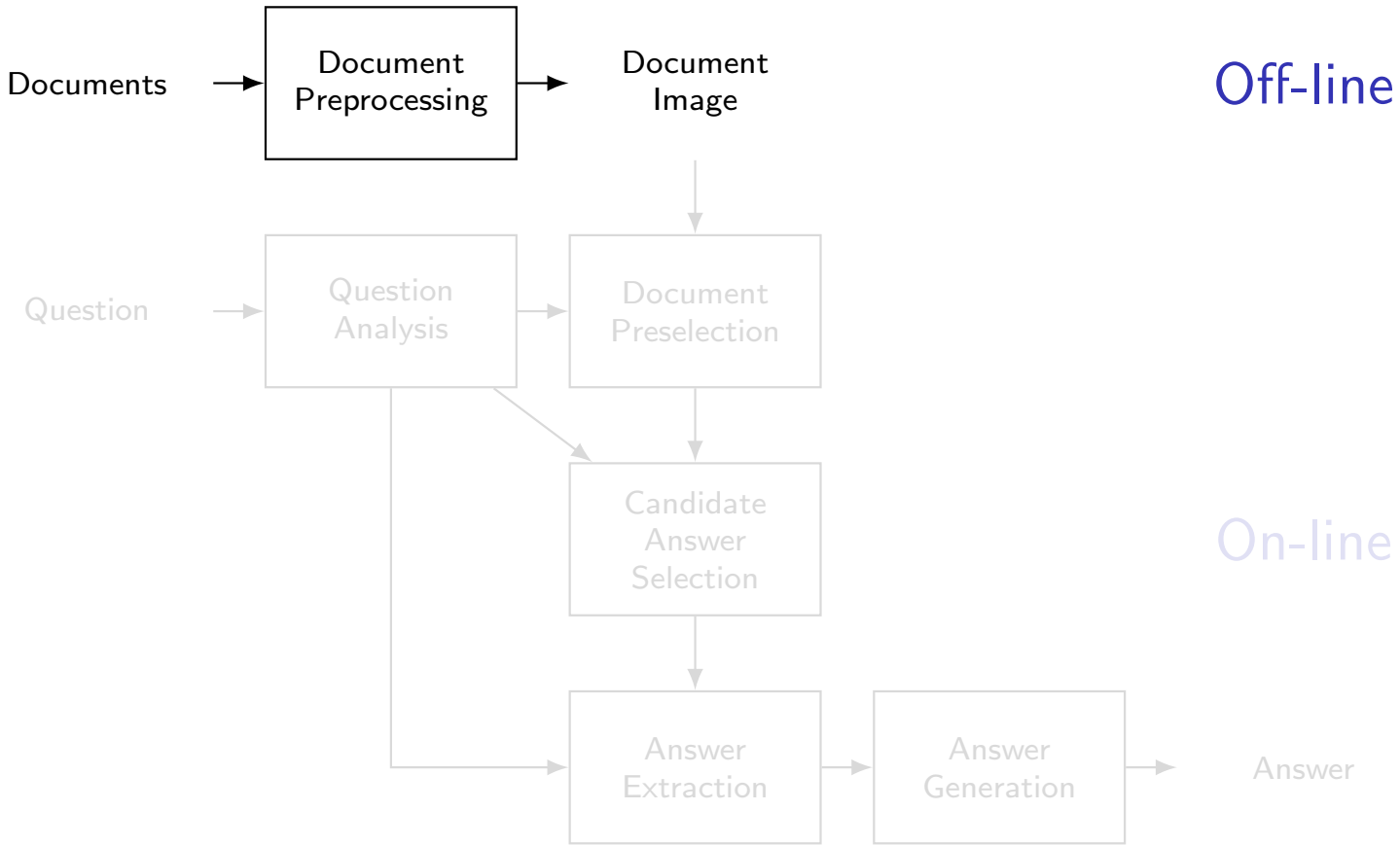
Issues in Question Answering

- Size of the corpus
- Types of questions
- Types of expected answers
- Paraphrases of questions and answers
- Locating the exact answer

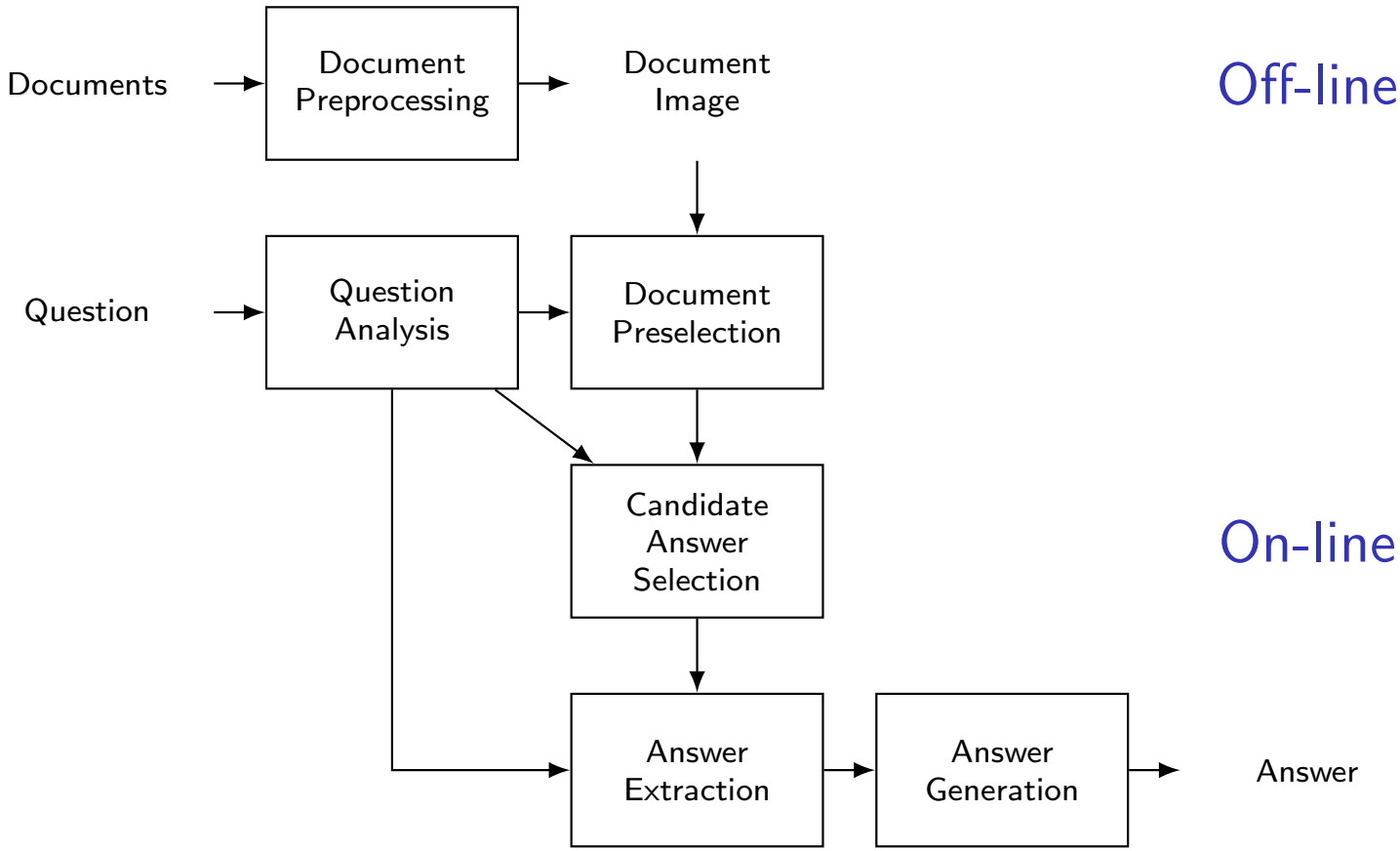
Issues in Question Answering

- Size of the corpus
- Types of questions
- Types of expected answers
- Paraphrases of questions and answers
- Locating the exact answer

Architecture



Architecture



Main Points of AnswerFinder I

Engineering

1. Build an environment for the development of QA systems
2. Emphasis on:

   Flexibility Easy to modify and adapt to diverse applications

   Configurability Easy to integrate new algorithms and to try different parameters
3. Integration of third-party modules whenever possible

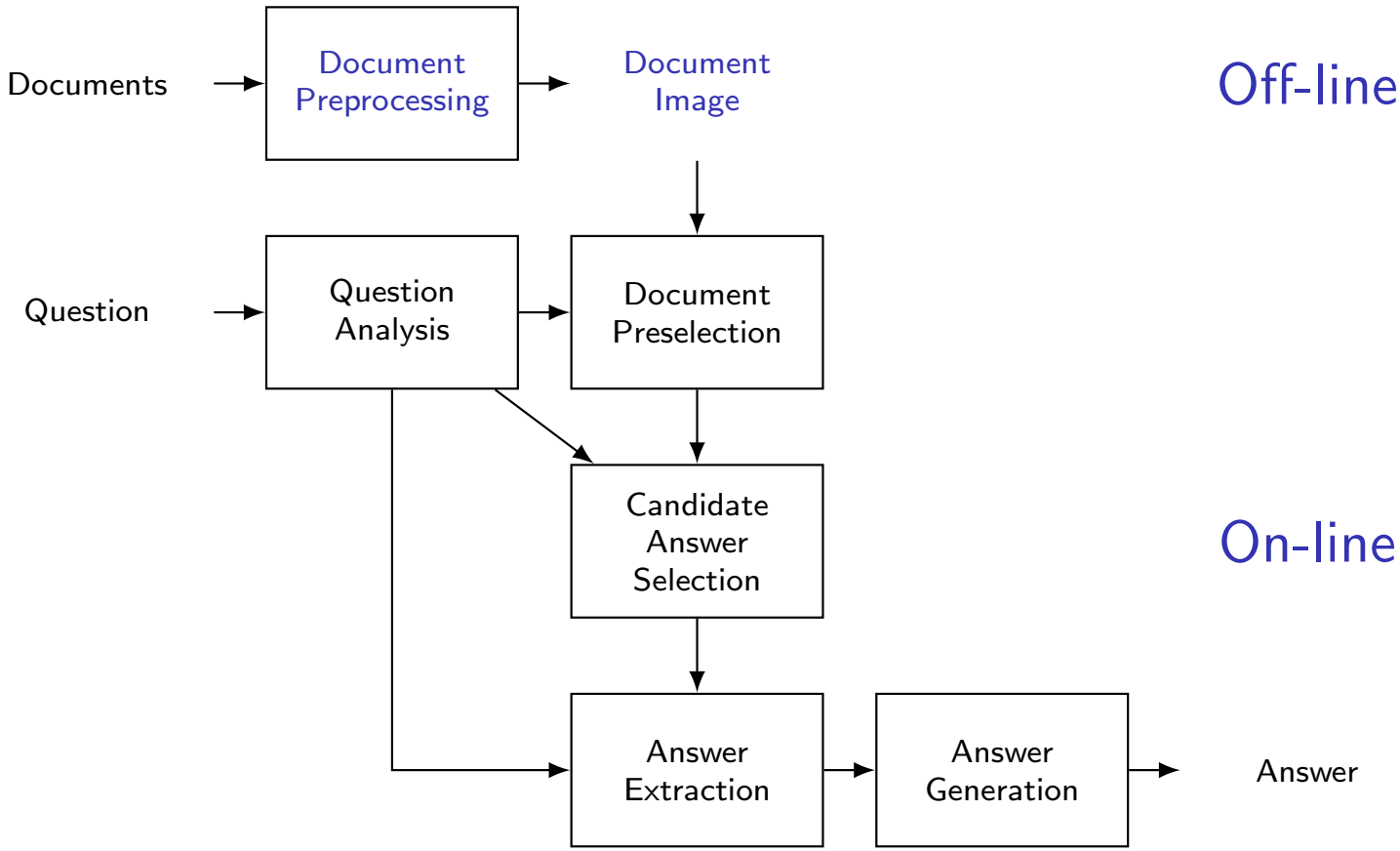
Main Points of AnswerFinder II

Theory

1. Focus on the selection of the answer sentence and extraction of the answer
   - But we are experimenting with the implementation of other modules as well
2. Use abstract (logical) representations of que question and text sentences
3. Integrate machine learning techniques
   - Learning of logical form patterns
   - Learning of weights in logical forms and patterns
   - Question classification (in the future)
   - Named entity recognition

Programme

Architecture



Document Preprocessing (Indexing)

Goals

1. Build a document image that can be used efficiently
2. Do as much as possible off-line

What we are Doing Here

1. Experimenting with using syntactic information and semantic labeling (named entities)

Document Preprocessing (Indexing)

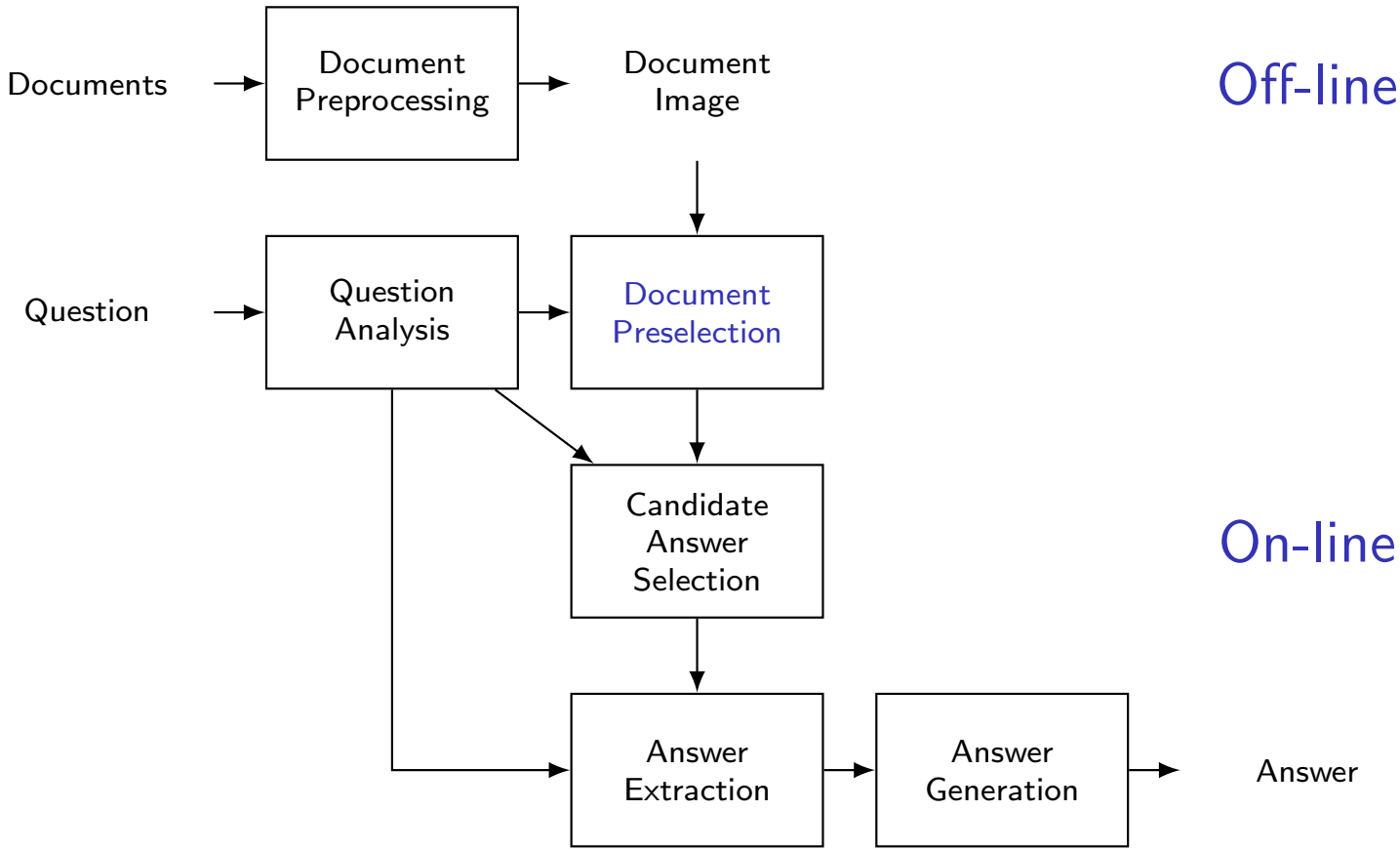
Goals

1. Build a document image that can be used efficiently
2. Do as much as possible off-line

What we are Doing Here

1. Experimenting with using syntactic information and semantic labeling (named entities)

Architecture



Document Preselection

Goal

Find the documents (or text fragments) with highest likelihood to contain the answer

Typical Methods

- Use an information retrieval system
  - As a blackbox, or
  - Adapted to the task, or
  - Designed specifically for the task

Document Preselection

Goal

Find the documents (or text fragments) with highest likelihood to contain the answer

Our Current Method

- Use an information retrieval system
  - As a blackbox, or
  - Adapted to the task, or
  - Designed specifically for the task

Document Preselection

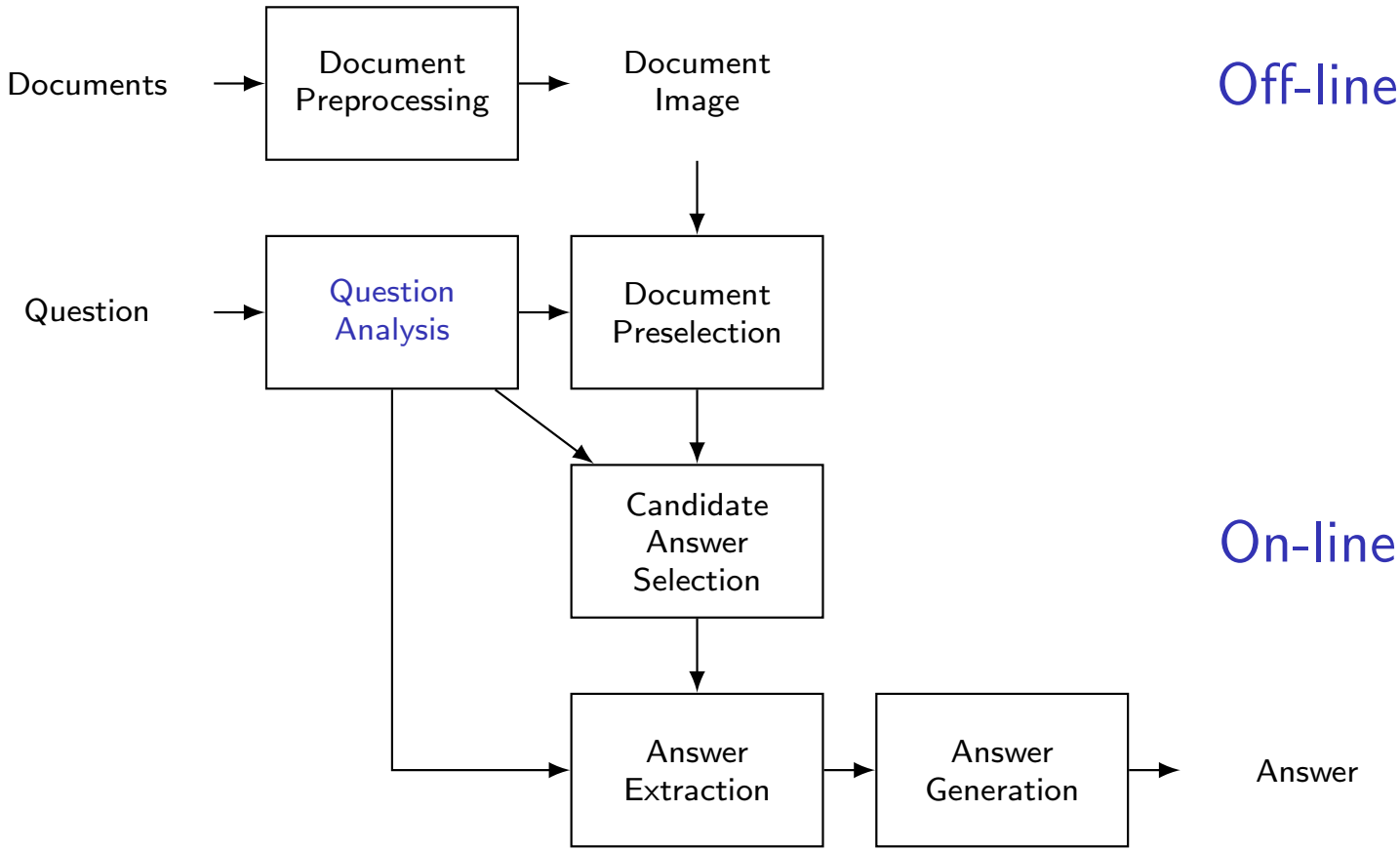
Goal

Find the documents (or text fragments) with highest likelihood to contain the answer

Our Future Method

- Use an information retrieval system
  - As a blackbox, or
  - Adapted to the task, or
  - Designed specifically for the task

Architecture



Question Analysis

Goal

Determine all the important information about the question

1. Question type
2. Expected answer type
3. Additional information (e.g. question focus)
4. Question representation
   - Keywords
   - Syntactic information
   - Semantic information

Our Method to Question Analysis

Current Method

- Use surface patterns of regular expressions

Coming Soon

- Learning by structure induction

Our Method to Question Analysis

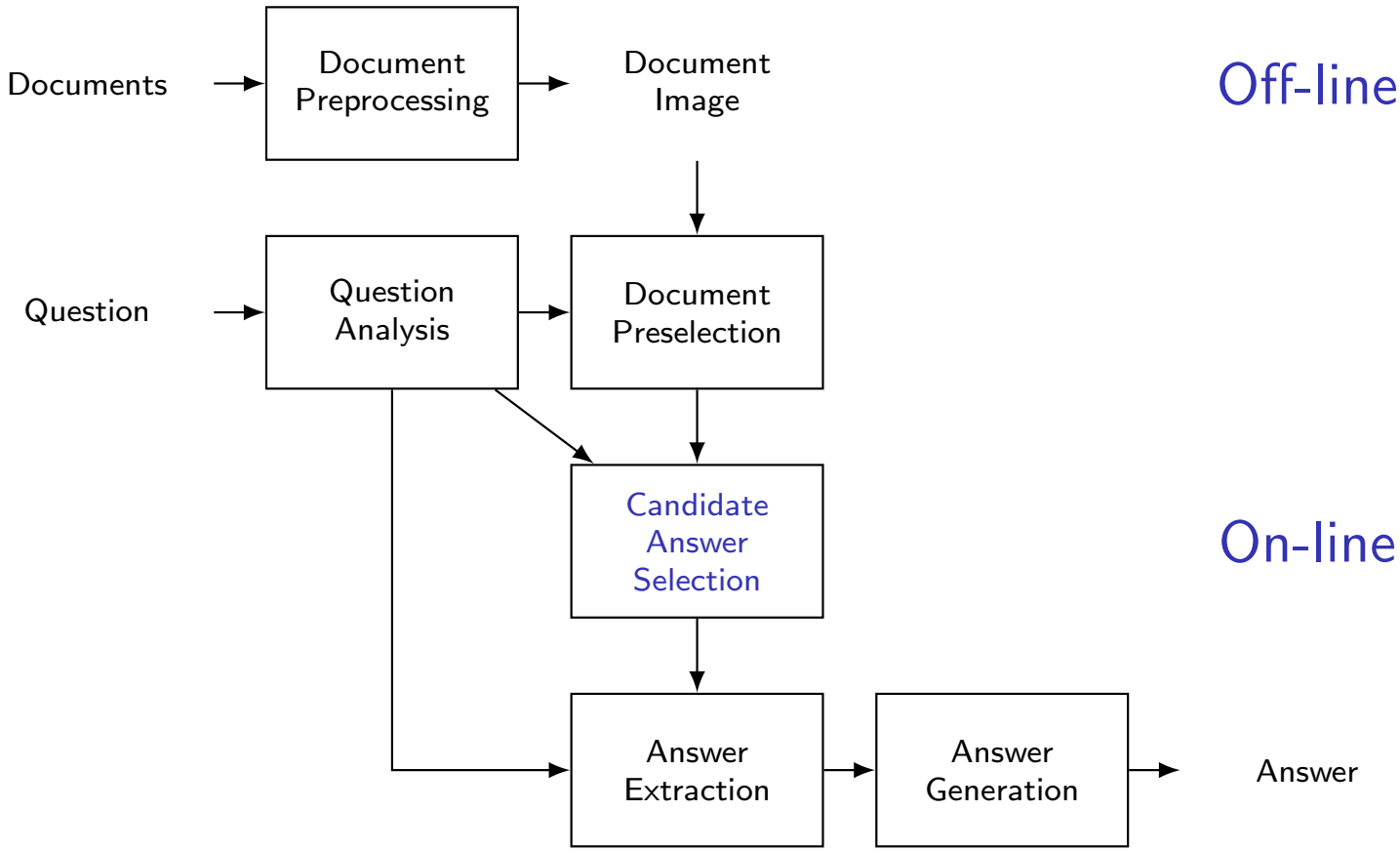
Current Method

- Use surface patterns of regular expressions

Coming Soon

- Learning by structure induction

Architecture



Candidate Answer Selection

Goal

Find all sentences (or text fragments) that contain the answer

Typical Method

1. Ignore all text zones that can't contain the answer
   - Use shallow methods
2. Find (and score) the sentences (fragments) containing the answer
   - Combination of shallow and deep methods

Candidate Answer Selection

Goal

Find all sentences (or text fragments) that contain the answer

Our Current Method

1. Ignore all sentences that do not contain a string of the expected answer type
   - Use a named entity recogniser
2. Score remaining sentences using a combination of lexical, syntactic, and semantic features

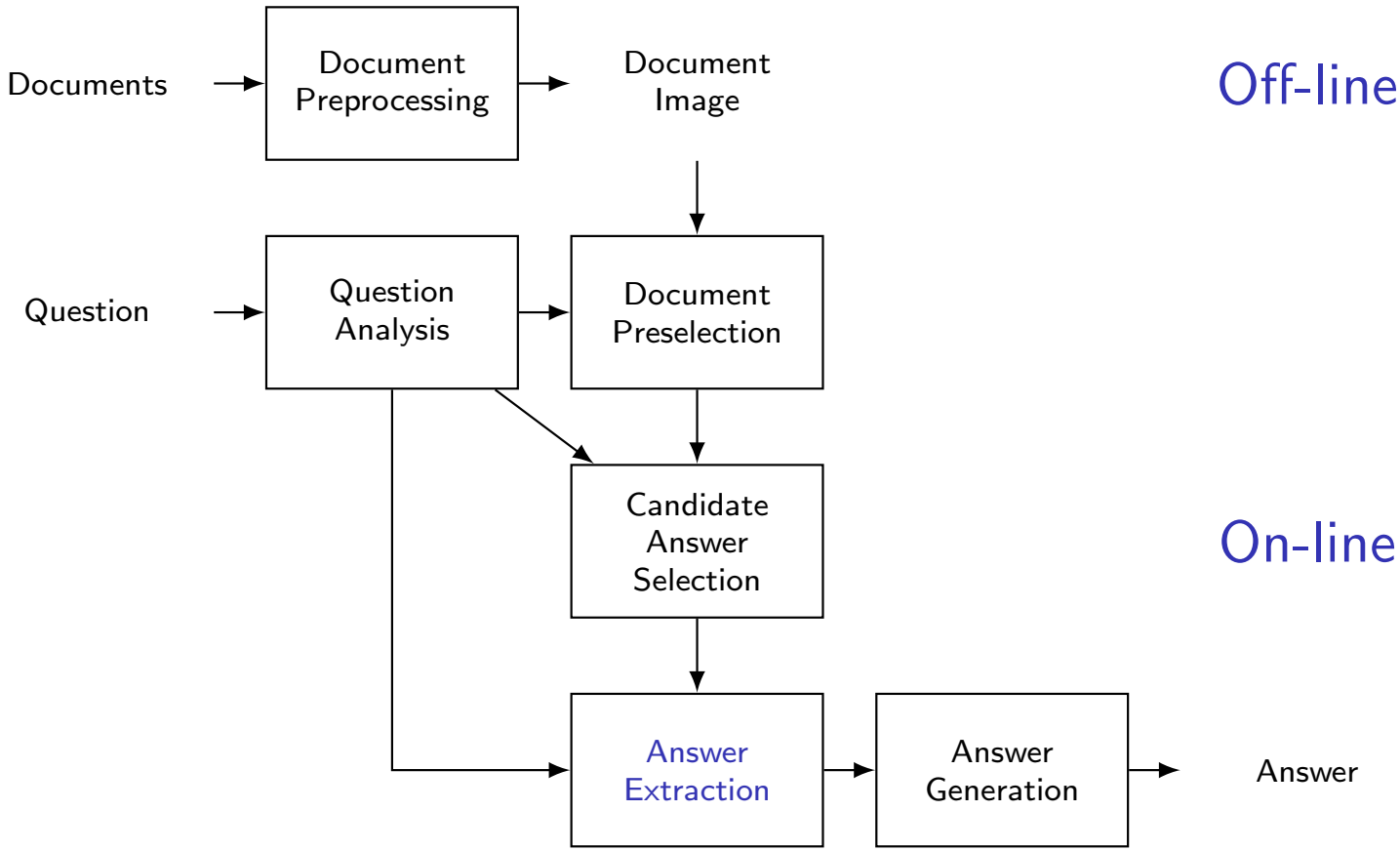
Our Sentence Scoring Method

Lexical Information Word overlap

Syntactical Information Grammatical relation overlap

Semantic Information Logical forms, logical graphs

Architecture



Answer Extraction

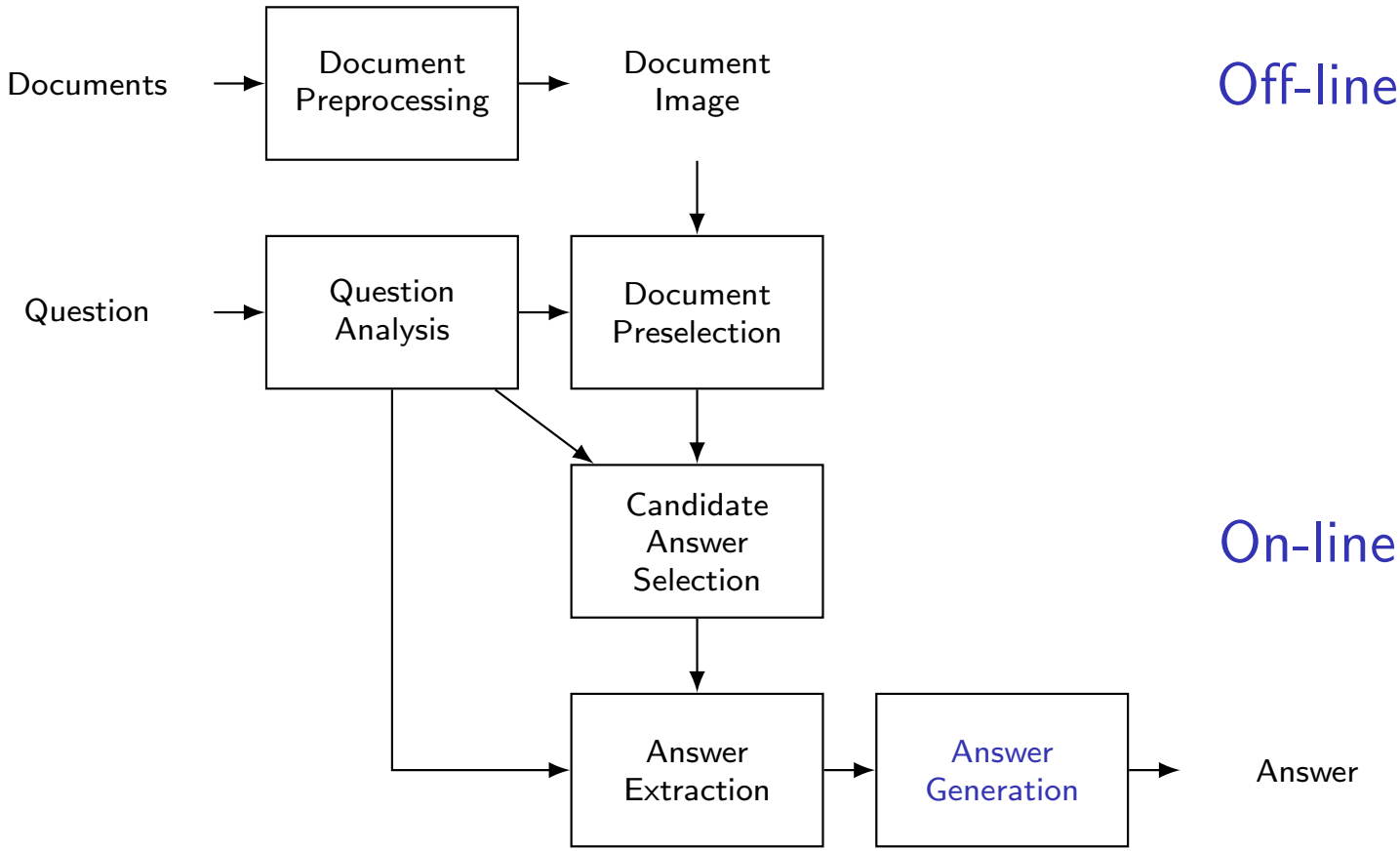
Goal

- Find the exact answer
- Elliminate noise

Typical Methods

- Use of lexical, syntactic, semantic information
- Use of patterns
- Answer validation
  - "Can I prove that this answer is not impossible?"
  - "Is there enough evidence that this is the answer?"

Architecture



Answer Generation

Goal

Present the answer to the user

Methods

- Present the exact answer
- Include links to the source
- Merge answers
- Adapt the answers to the user model

Programme

Background Info

Rule-based Approaches to QA

- Find answers by applying rules ▸ example
- Need to write a large number of rules
  - The problem of paraphrasing
- Problems with portability

So let's learn the QA rules

Background Info

Rule-based Approaches to QA

- Find answers by applying rules ▸ example
- Need to write a large number of rules
  - The problem of paraphrasing
- Problems with portability

So let's learn the QA rules

Example of a Rule

Question Pattern

Who is the <X>$_{position}$ of <Y>$_{country}$

Answer Pattern

Y's X ANSWER$_{two\ capitalised\ words}$

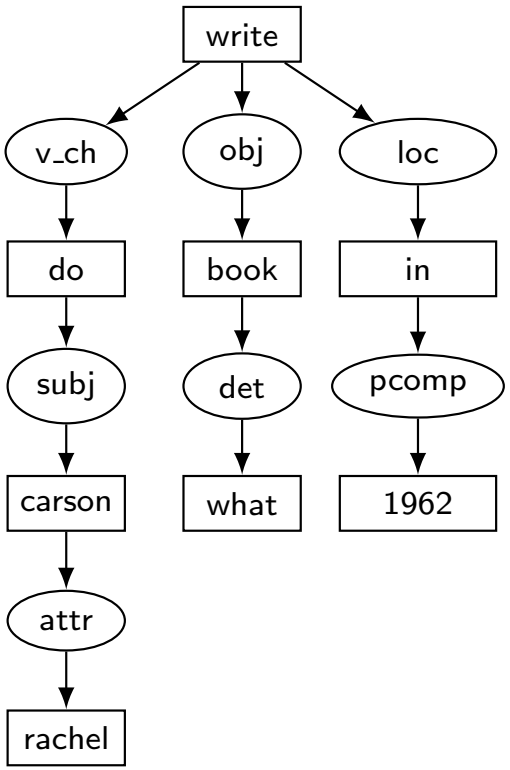
The Rule Applied

Q *Who is the <Prime Minister>$_{position}$ of <Australia>$_{country}$?*

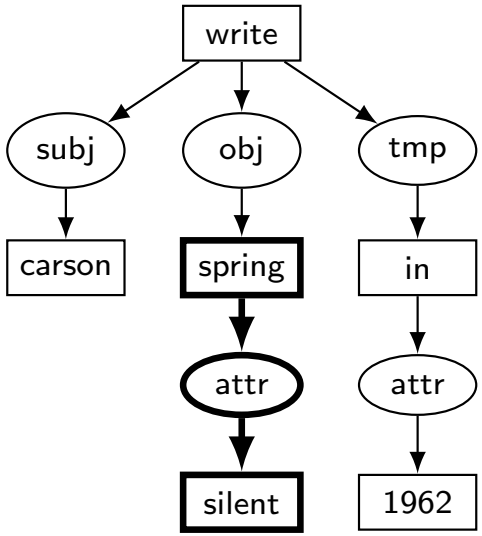
A *<Australia>'s <Prime Minister> <John Howard>$_{ANSWER}$*

Graph Rule from a Question Answering Pair

Q: *What book did Rachel Carson write in 1962?*

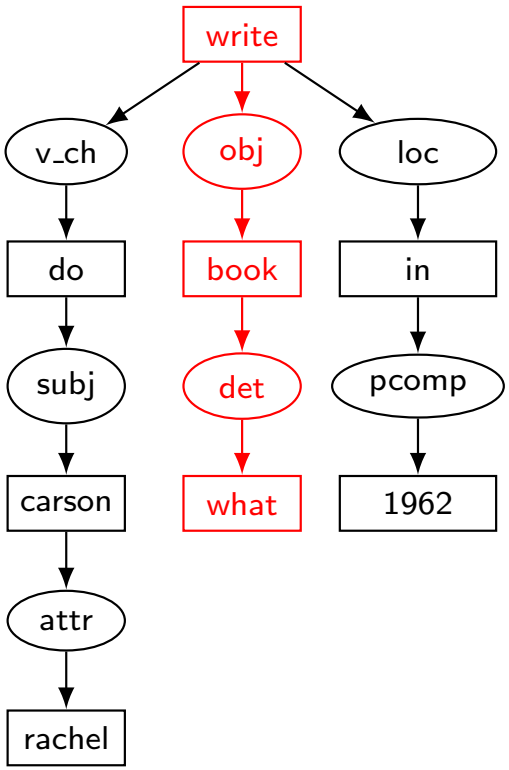


A: *In 1962 Carson wrote* **"Silent Spring"**

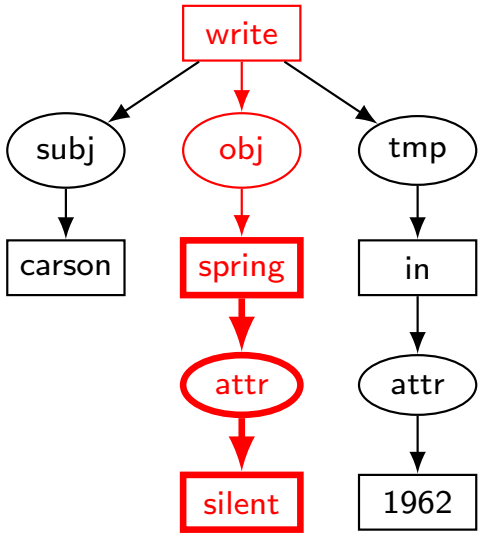


Graph Rule from a Question Answering Pair

Q: *What book did Rachel Carson write in 1962?*

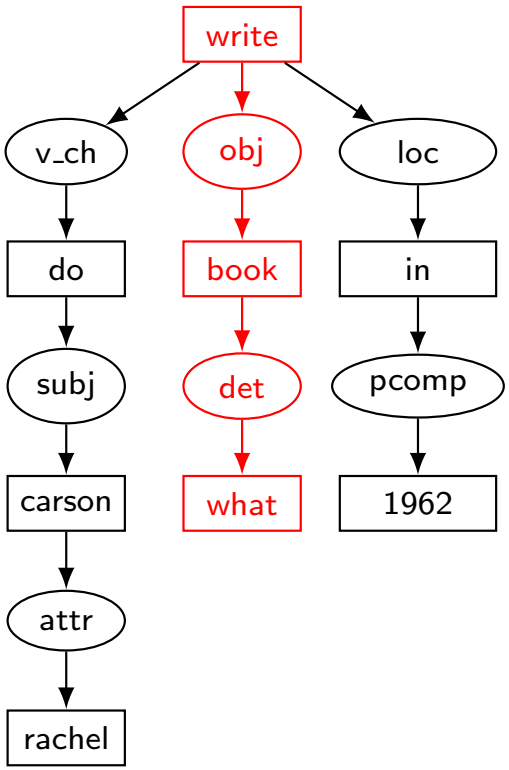


A: *In 1962 Carson wrote* **"Silent Spring"**

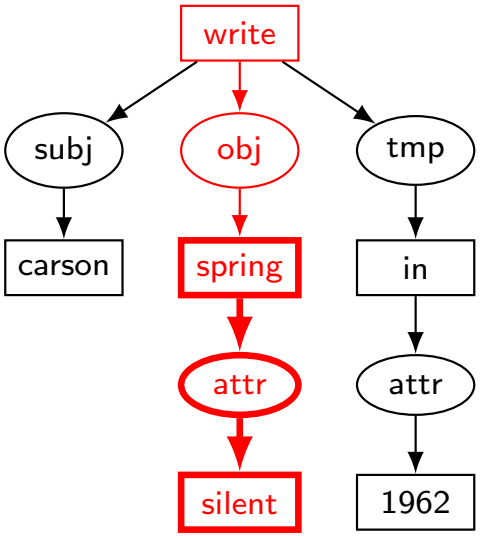


Graph Rule from a Question Answering Pair

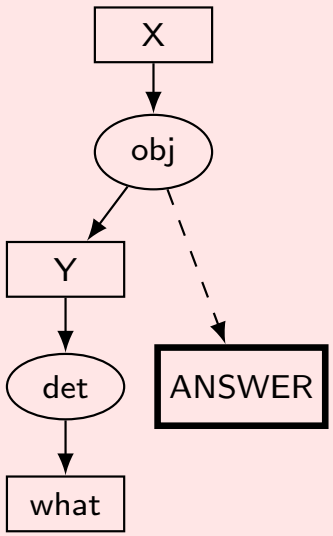
Q: *What book did Rachel Carson write in 1962?*



A: *In 1962 Carson wrote* **"Silent Spring"**

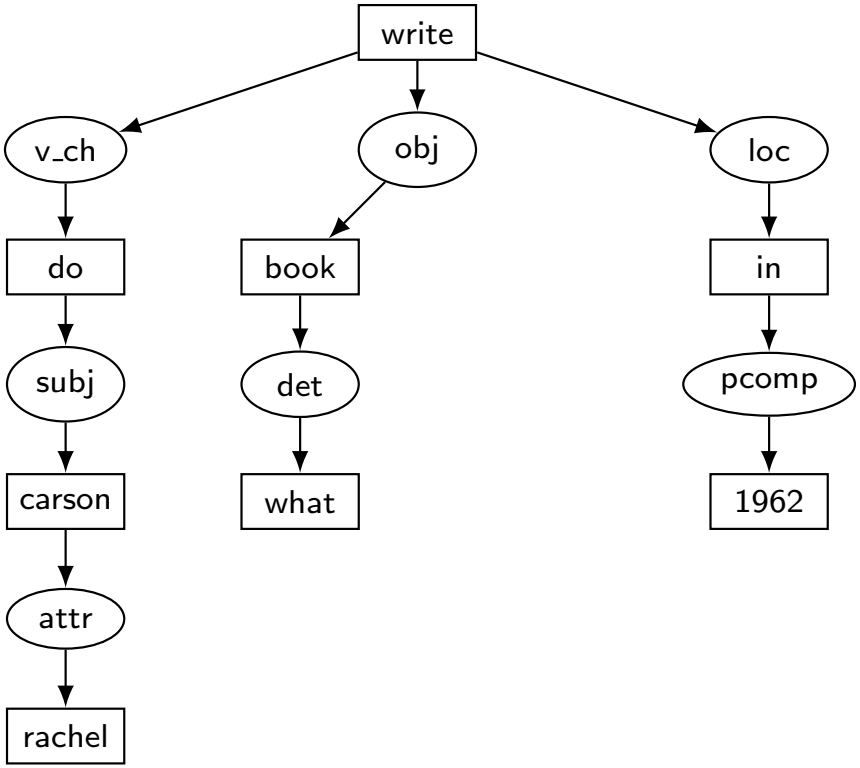


The Rule

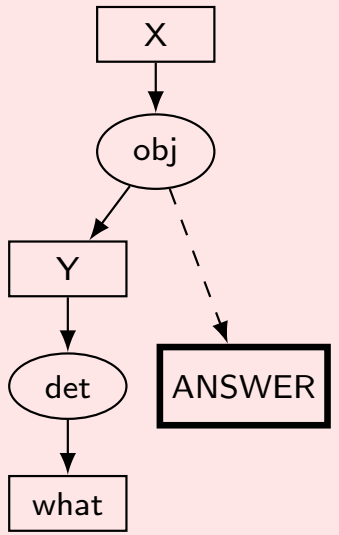


Applying the Rule to the Question

Q: *What book did Rachel Carson write in 1962?*

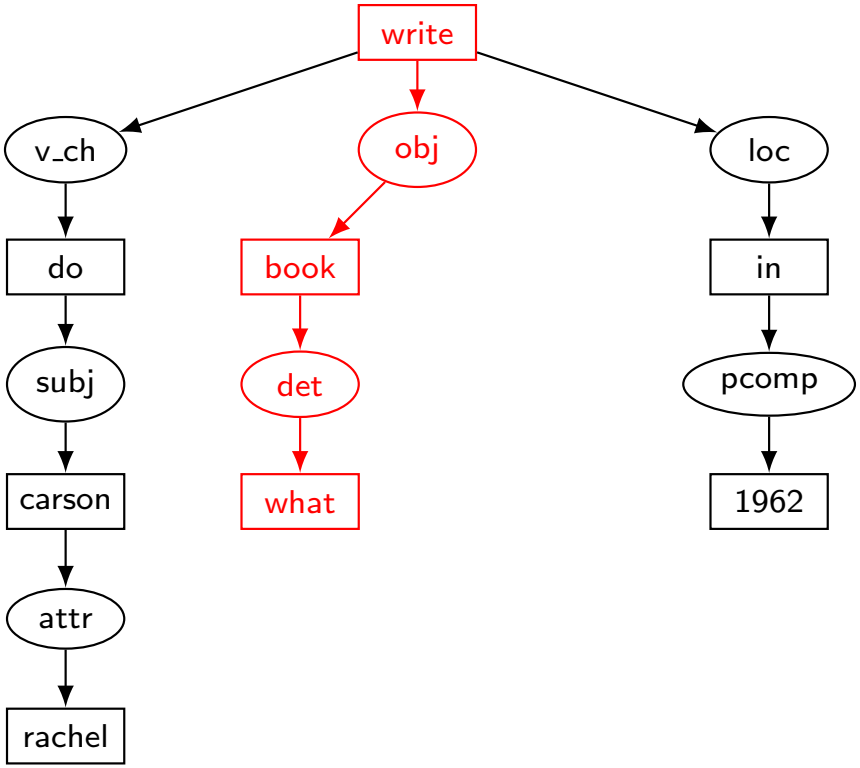


The Rule

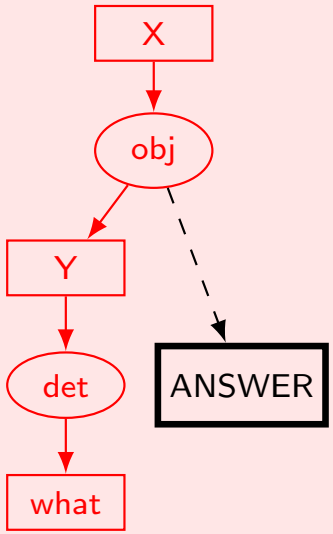


Applying the Rule to the Question

Q: *What book did Rachel Carson write in 1962?*

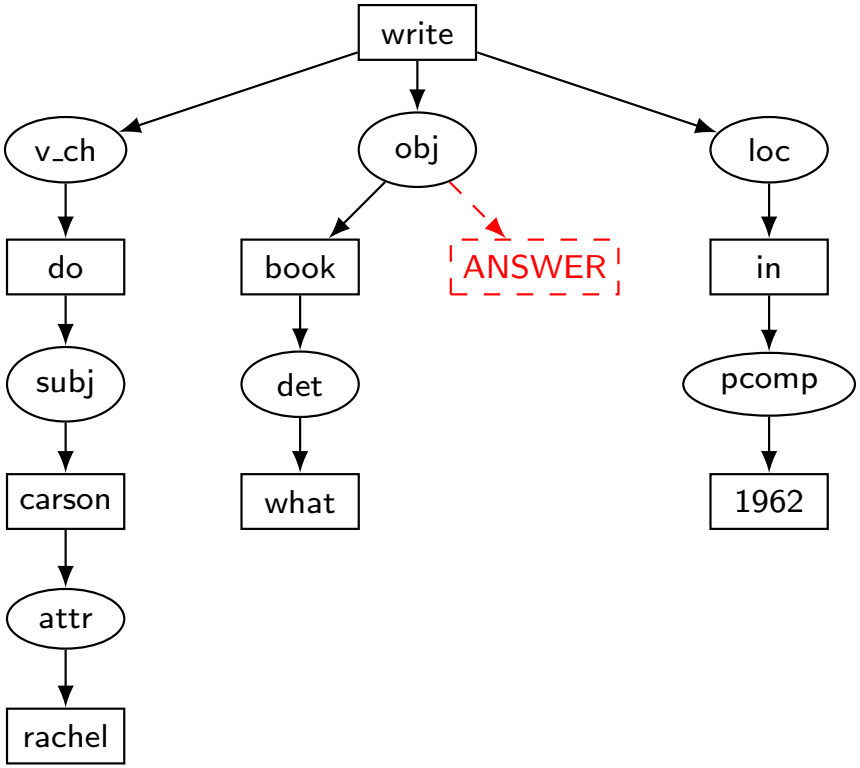


The Rule

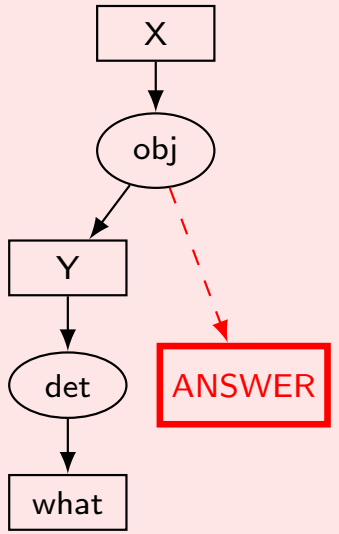


Applying the Rule to the Question

Q: *What book did Rachel Carson write in 1962?*

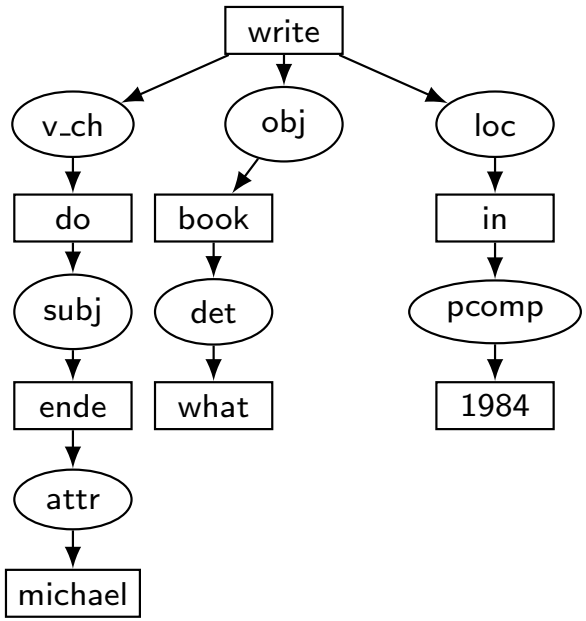


The Rule

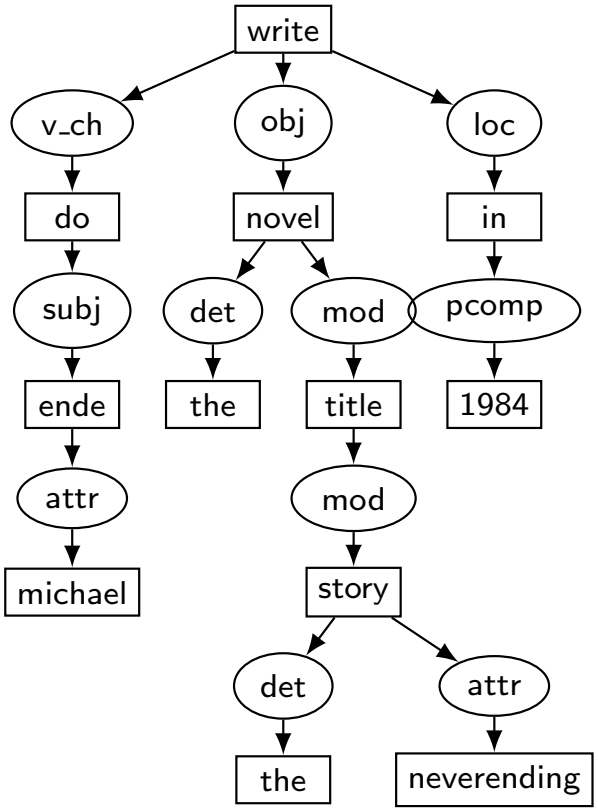


Question Answering

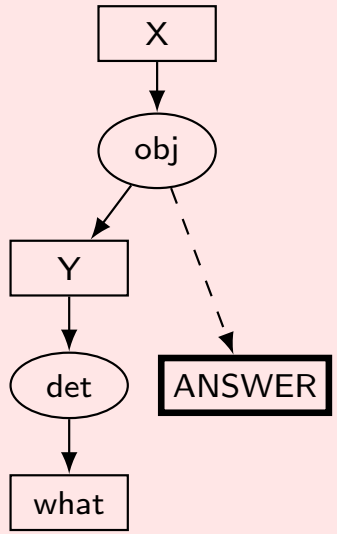
Q: *What book did Michael Ende write in 1984?*



A: *In 1984 Michael Ende wrote the novel titled "The Neverending Story"*

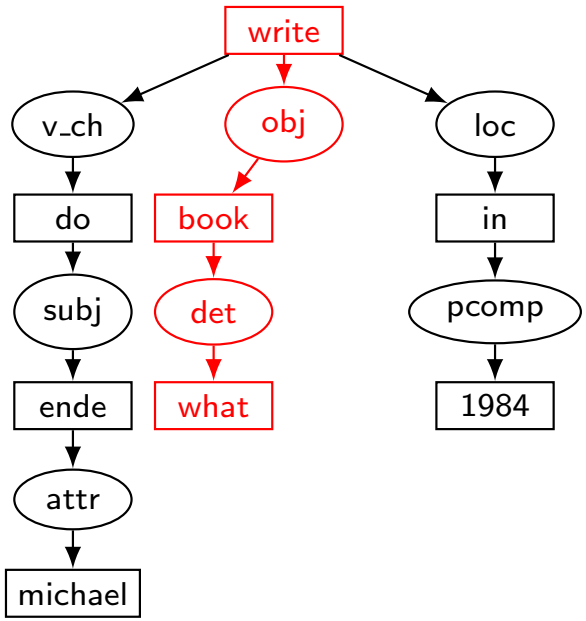


The Rule

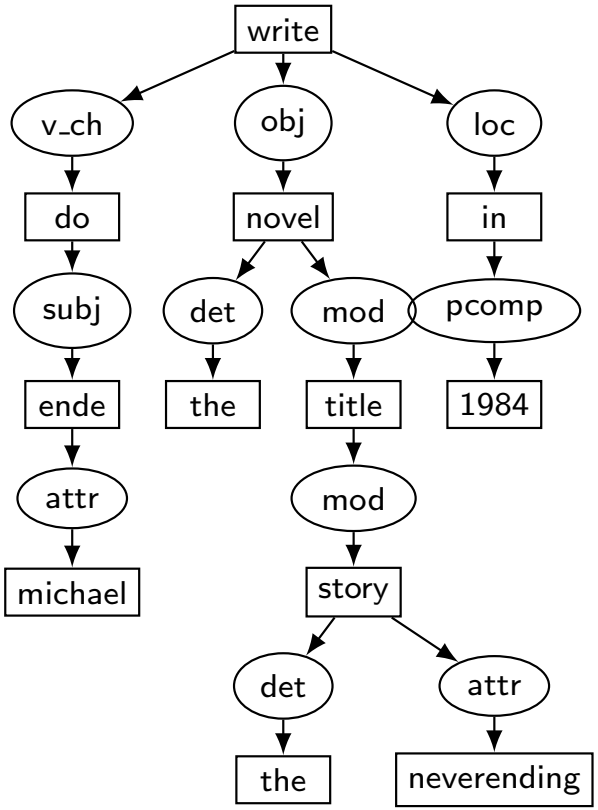


Question Answering

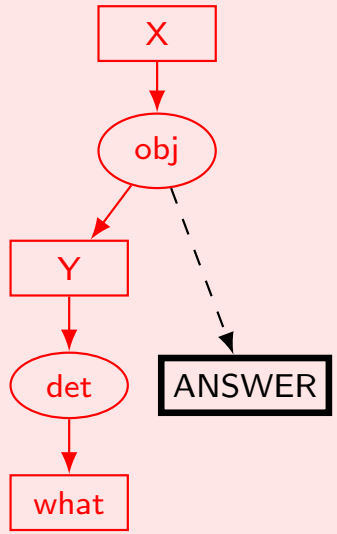
Q: *What book did Michael Ende write in 1984?*



A: *In 1984 Michael Ende wrote the novel titled "The Neverending Story"*

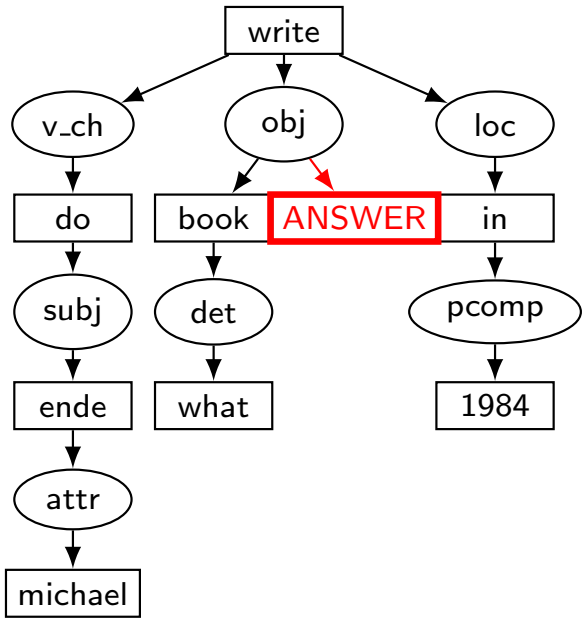


The Rule

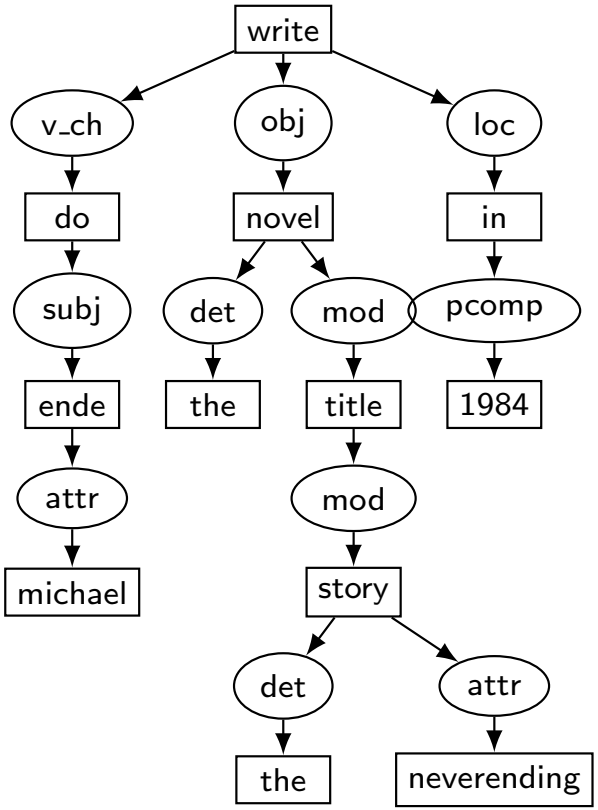


Question Answering

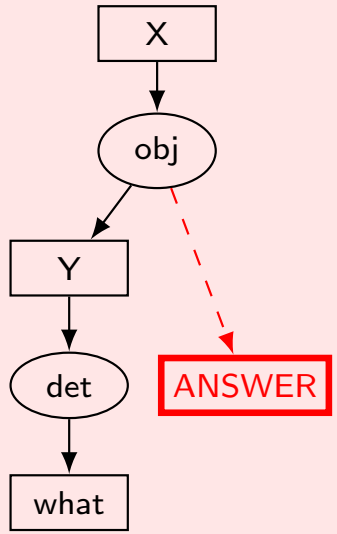
Q: *What book did Michael Ende write in 1984?*



A: *In 1984 Michael Ende wrote the novel titled "The Neverending Story"*

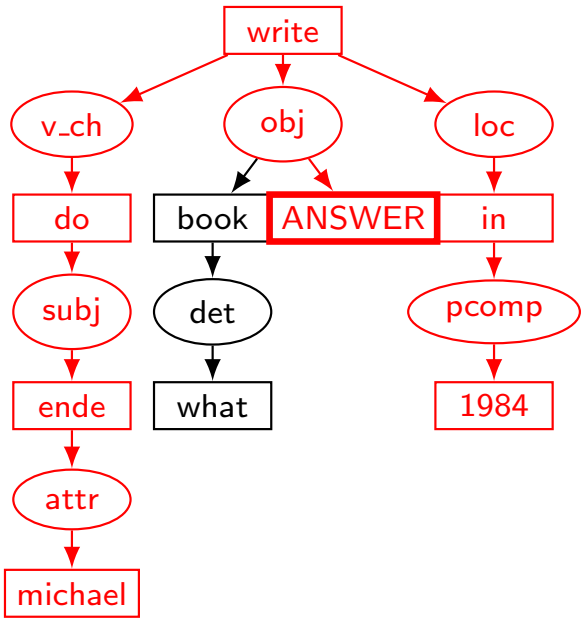


The Rule

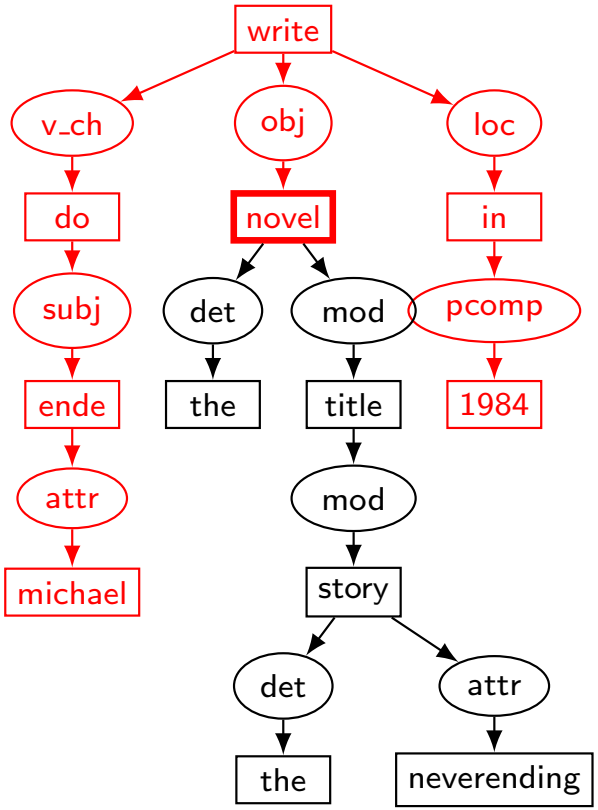


Question Answering

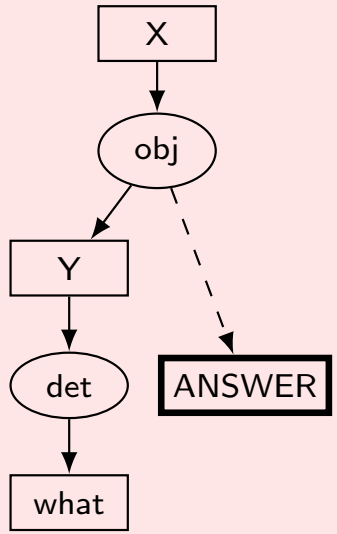
Q: *What book did Michael Ende write in 1984?*



A: *In 1984 Michael Ende wrote the novel titled "The Neverending Story"*

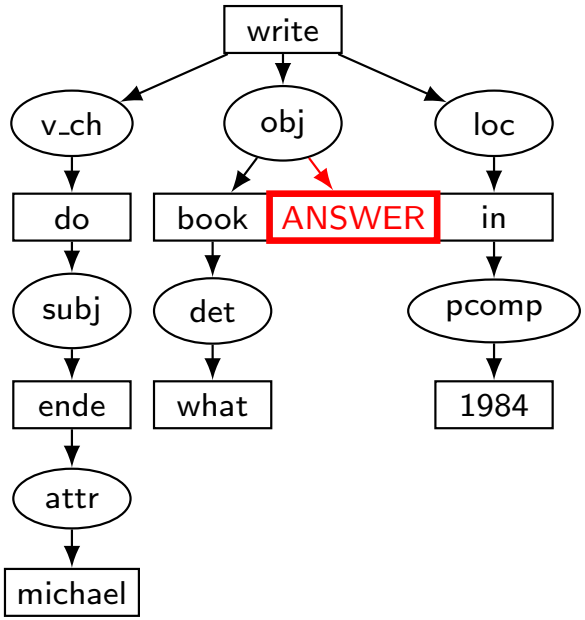


The Rule

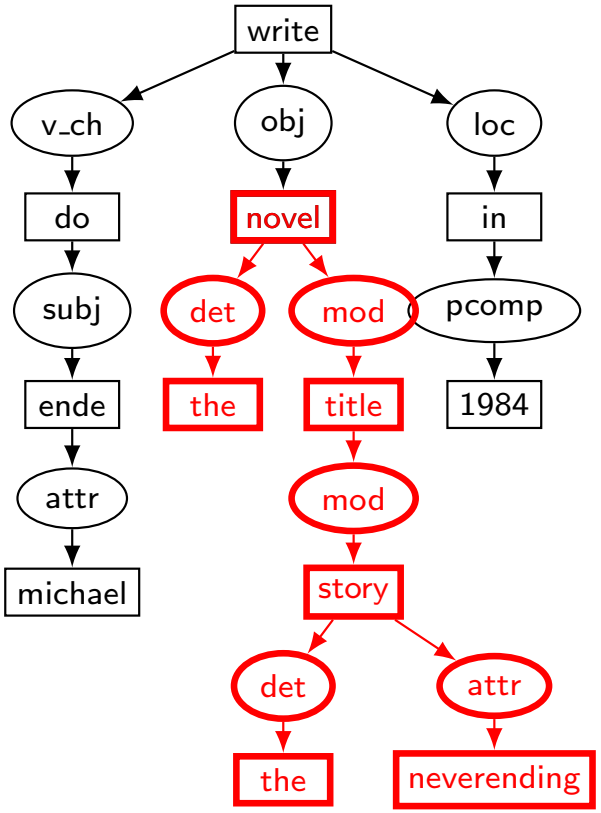


Question Answering

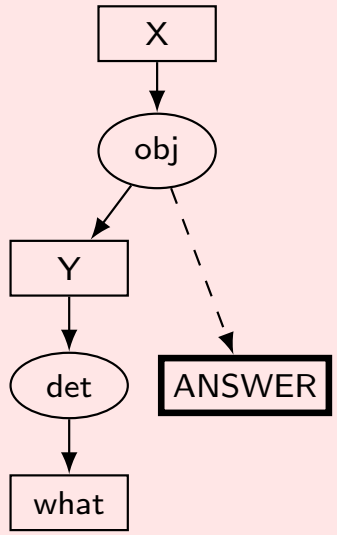
Q: *What book did Michael Ende write in 1984?*



A: *In 1984 Michael Ende wrote the novel titled "The Neverending Story"*



The Rule



Programme

The Method

Finding the Question Pattern

- Question pattern: overlap between training question and answer sentence
  - Thus, question-only terms are ignored
- Graph Theory: Find the maximum common subgraph

Finding the Extension Graph

- Extension graph: path from the MCS to the answer:
  - The subgraph of the answer sentence that connects the MCS with the answer

The Method

Finding the Question Pattern

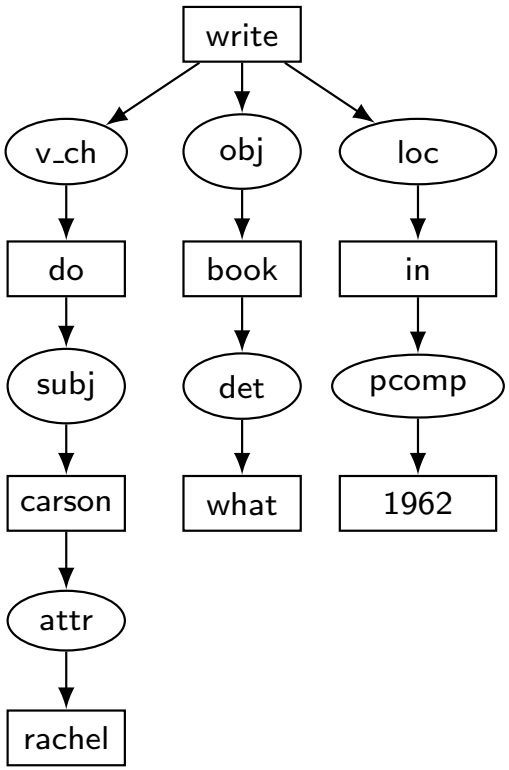
- Question pattern: overlap between training question and answer sentence
  - Thus, question-only terms are ignored
- Graph Theory: Find the maximum common subgraph

Finding the Extension Graph

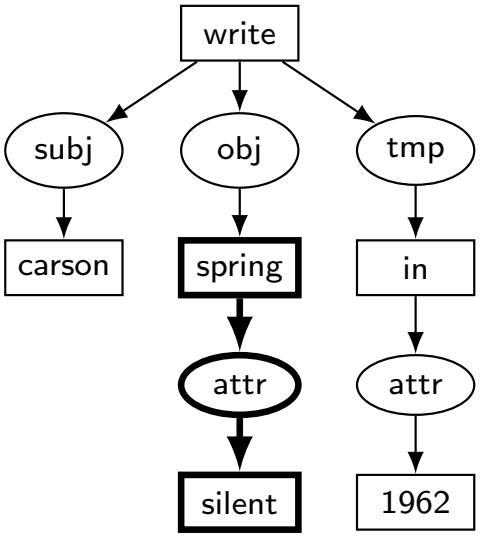
- Extension graph: path from the MCS to the answer:
  - The subgraph of the answer sentence that connects the MCS with the answer

Example

Q: *What book did Rachel Carson write in 1962?*



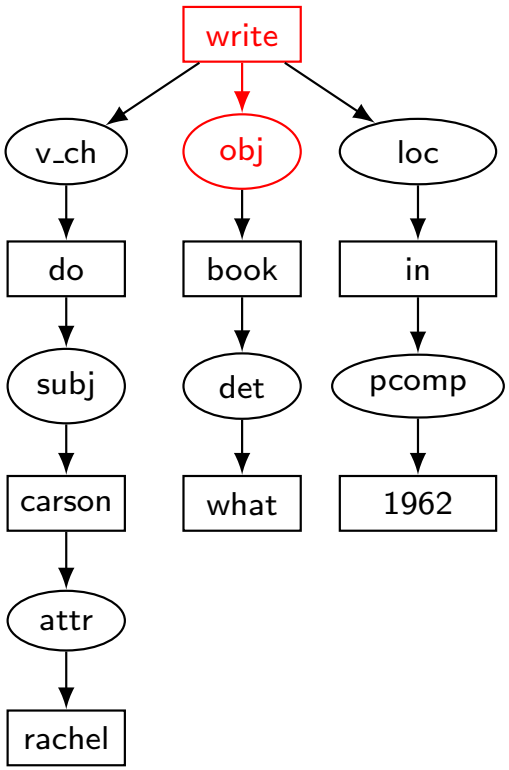
A: *In 1962 Carson wrote* **"Silent Spring"**



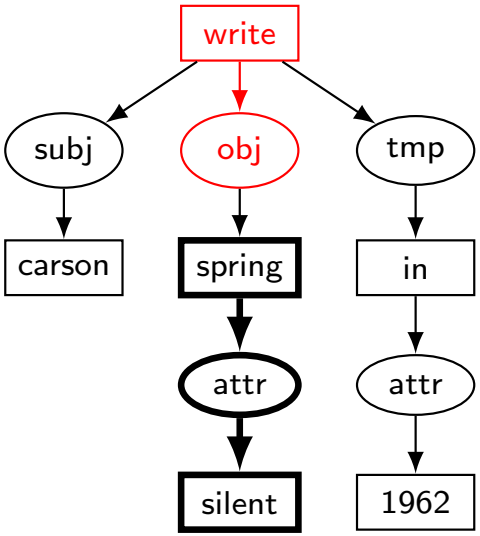
The Rule

Example

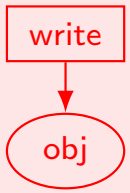
Q: *What book did Rachel Carson write in 1962?*



A: *In 1962 Carson wrote* **"Silent Spring"**

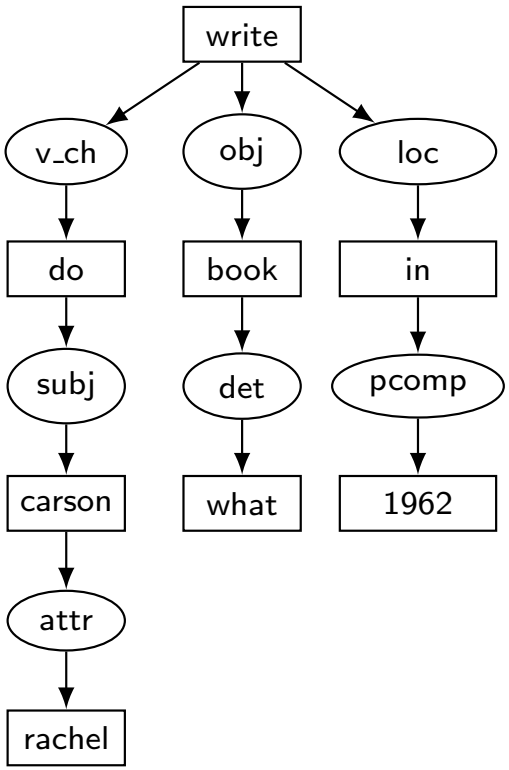


The Rule

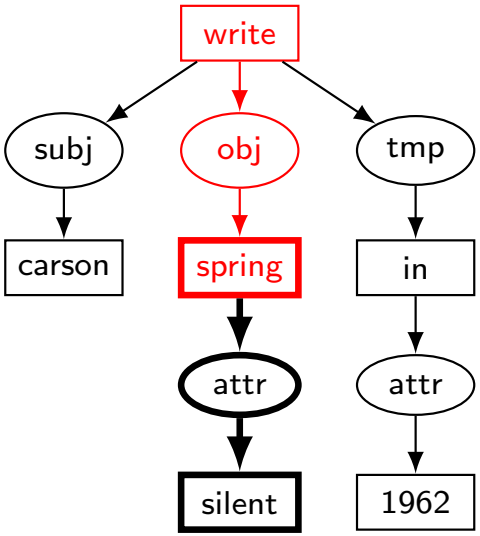


Example

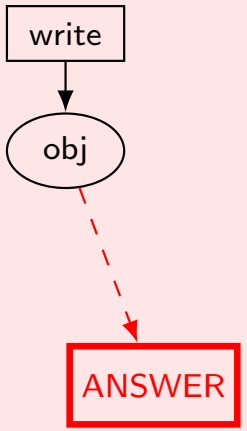
Q: *What book did Rachel Carson write in 1962?*



A: *In 1962 Carson wrote* **“Silent Spring”**



The Rule



Generalising and Weighting the Rule

Generalising

- The rule needs to be generalised to make it apply to unknown data

Weighting

- Some rules are better than others; need to weight them
- Weight based on rule precision on training data:

$$\mathcal{W}(r) = \frac{\#\text{ correct answers found}}{\#\text{ answers found}}$$

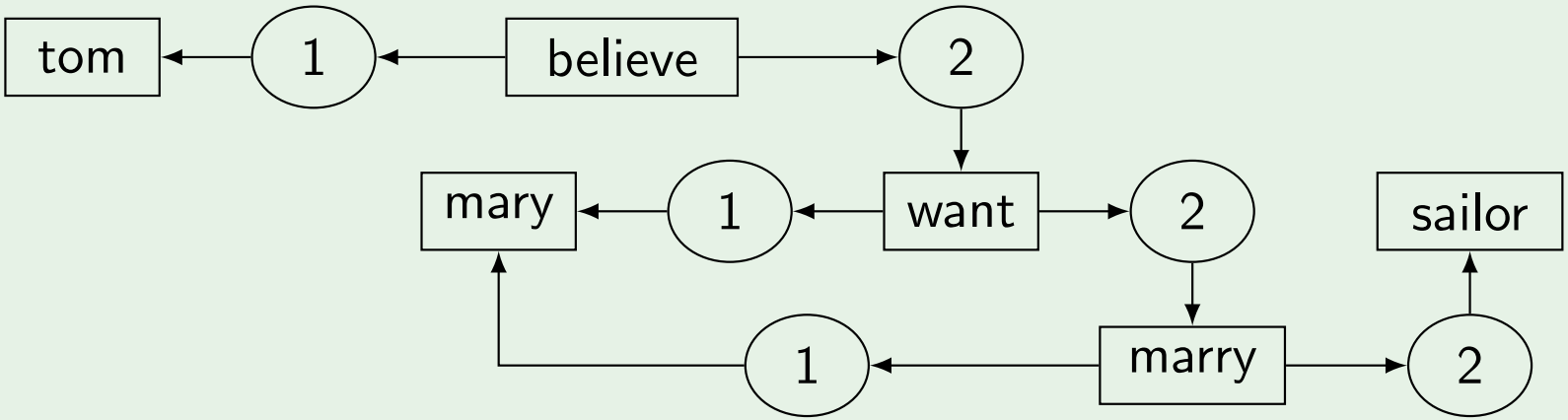
Programme

A Logical Graph

Features

- Format inspired on Sowa's Conceptual Graphs
- Directed, bipartite graphs
  - Concepts
  - Relations
- Built from the output of Connexor FDG parser

Tom believes that Mary wants to marry a sailor



QA with Logical Graphs I

Generalising the Graph Rule

- Relations are left unmodified
- Concepts
  are converted into variables, except for a list of "stop concepts"

  and, or, not, nor, if, otherwise, have, be, become, do, make

QA with Logical Graphs II

Ranking the Answer

- Answer score: overlap size $\times$ rule weight
- Overlap size: weighted sum of concepts and relations
  - Concept and relation weights inspired on IDF (Inverse document frequency)
- Answer duplicates merge by adding up their individual scores
- Answers occurring in the question are ignored

Further Work

- Implement more accurate generalisations
  - Combine WordNet with a NE recogniser
  - Use ML techniques (grammar induction) to learn types of vertices
- Introduce question-only words in the question pattern
- Pay further attention to paraphrases
  - Synonyms
  - Nominalisations
  - Hyponyms

Thank You!

Any questions...?