

Expert Systems

Week 3: Demonstration rule-based expert system

Dr. Mohammed A. Altahrawi

University College of Applied Sciences

November 16, 2024

- 1 CORE CONCEPTS OF EXPERTA
- 2 STEPS TO IMPLEMENT EXPERTA
- 3 DEMONSTRATION OF RULE-BASED (TRAFFIC LIGHT DECISION SYSTEM)
- 4 DEMONSTRATION OF RULE-BASED MEDIA-ADVISOR
- 5 CONFLICT RESOLUTION
 - conflict resolution methods
- 6 METAKNOWLEDGE
- 7 ASSIGNMENT

- 1 CORE CONCEPTS OF EXPERTA
- 2 STEPS TO IMPLEMENT EXPERTA
- 3 DEMONSTRATION OF RULE-BASED (TRAFFIC LIGHT DECISION SYSTEM)
- 4 DEMONSTRATION OF RULE-BASED MEDIA-ADVISOR
- 5 CONFLICT RESOLUTION
 - conflict resolution methods
- 6 METAKNOWLEDGE
- 7 ASSIGNMENT

Core Concepts of Experta

- Experta is a forward-chaining rule engine, meaning it derives conclusions (new facts) by applying a series of rules to existing facts.
- The process relies on several key concepts:
 - **Facts:** These are statements or pieces of information that the system knows, such as "traffic light is red" or "cars are waiting." Facts serve as inputs and outputs of the system.
 - **Rules:** Rules define conditional logic that acts on facts. A rule can check the values of specific facts and then trigger actions or declare new facts based on the conditions.
 - **Knowledge Engine:** The core engine of Experta is the KnowledgeEngine class, which manages facts and rules. When the engine runs, it evaluates the facts and triggers rules as applicable.
- **Forward-Chaining:** experta uses forward-chaining, where rules are triggered by current facts. When a rule is triggered, it may declare new facts, which in turn can trigger additional rules.

Outlines

- 1 CORE CONCEPTS OF EXPERTA
- 2 STEPS TO IMPLEMENT EXPERTA
- 3 DEMONSTRATION OF RULE-BASED (TRAFFIC LIGHT DECISION SYSTEM)
- 4 DEMONSTRATION OF RULE-BASED MEDIA-ADVISOR
- 5 CONFLICT RESOLUTION
 - conflict resolution methods
- 6 METAKNOWLEDGE
- 7 ASSIGNMENT

Steps to Implement Experta

- **Define Fact Classes:** These represent different types of information (facts) the system can handle.
- **Create Rules with @Rule Decorator:** Each rule specifies conditions under which it should be triggered. When the conditions are met, the rule's method is executed.
- **Initialize and Run the Knowledge Engine:** After defining the facts and rules, instantiate the KnowledgeEngine, add initial facts, and run it

- 1 CORE CONCEPTS OF EXPERTA
- 2 STEPS TO IMPLEMENT EXPERTA
- 3 DEMONSTRATION OF RULE-BASED (TRAFFIC LIGHT DECISION SYSTEM)
- 4 DEMONSTRATION OF RULE-BASED MEDIA-ADVISOR
- 5 CONFLICT RESOLUTION
 - conflict resolution methods
- 6 METAKNOWLEDGE
- 7 ASSIGNMENT

Demonstration of rule-based (Traffic Light Decision System)

- **Step 1:** Import Experta and Define Facts.

```
from experta import Fact, Rule, KnowledgeEngine
```

- **Step 2:** Define fact classes for the traffic light color and car action.

```
class TrafficLight(Fact):  
    """Represents the color of the traffic light."""  
    pass  
class CarAction(Fact):  
    """Represents the action the car should take."""  
    pass
```

- **Step 3:** Define the Knowledge Engine with Rules: create a **TrafficControl** class that inherits from **KnowledgeEngine**.
- This class will hold the rules for determining the car's action based on the traffic light color.

Python Code Example

```
class TrafficControl(KnowledgeEngine):

    @Rule(TrafficLight(color='red'))
    def stop(self):
        print("Rule triggered: Light is red. Car
              should STOP.")
        self.declare(CarAction(action='stop'))

    @Rule(TrafficLight(color='yellow'))
    def slow_down(self):
        print("Rule triggered: Light is yellow. Car
              should SLOW DOWN.")
        self.declare(CarAction(action='slow down'))

    @Rule(TrafficLight(color='green'))
    def go(self):
        print("Rule triggered: Light is green. Car
              should GO.")
        self.declare(CarAction(action='go'))
```

- `traffic_control.reset()`: Resets and initializes the knowledge engine, clearing previous facts or states.
- `traffic_control.declare(TrafficLight(color='red'))`: Declares an initial fact that the traffic light is red.
- `traffic_control.run()`: Starts the forward-chaining process. The engine evaluates rules based on the declared facts and triggers applicable rules.

snappiest from the code results

```
▶ # Change the color to yellow and run the engine again
traffic_control.reset()
traffic_control.declare(TrafficLight(color='yellow'))
traffic_control.run()

# Change the color to green and run the engine again
traffic_control.reset()
traffic_control.declare(TrafficLight(color='green'))
traffic_control.run()
```

⇌ Rule triggered: Light is yellow. Car should SLOW DOWN.
Rule triggered: Light is green. Car should GO.

Figure: Traffic light_example_code

Outlines

- 1 CORE CONCEPTS OF EXPERTA
- 2 STEPS TO IMPLEMENT EXPERTA
- 3 DEMONSTRATION OF RULE-BASED (TRAFFIC LIGHT DECISION SYSTEM)
- 4 DEMONSTRATION OF RULE-BASED MEDIA-ADVISOR
- 5 CONFLICT RESOLUTION
 - conflict resolution methods
- 6 METAKNOWLEDGE
- 7 ASSIGNMENT

Demonstration of rule-based Media-Advisor

- The Leonardo expert system shell was selected as a tool to build a decision-support system called MEDIA ADVISOR.
- The system provides advice on selecting a medium for delivering a training program based on the **trainee's job**.

Example

if a trainee is a mechanical technician responsible for maintaining hydraulic systems, an appropriate medium might be a workshop, where the trainee could learn how basic hydraulic components operate, how to troubleshoot hydraulics problems and how to make simple repairs to hydraulic systems.

Media-Advisor Example

Rule: 1

if the environment is papers
or the environment is manuals
or the environment is documents
or the environment is textbooks
then the stimulus_situation is verbal

Rule: 2

if the environment is pictures
or the environment is illustrations
or the environment is photographs
or the environment is diagrams
then the stimulus_situation is visual

Rule: 3

if the environment is machines
or the environment is buildings
or the environment is tools
then the stimulus_situation is 'physical object'

Rule: 4

if the environment is numbers
or the environment is formulas
or the environment is 'computer programs'
then the stimulus_situation is symbolic

Rule: 5

if the job is lecturing
or the job is advising
or the job is counselling
then the stimulus_response is oral

Rule: 6

if the job is building
or the job is repairing
or the job is troubleshooting
then the stimulus_response is 'hands-on'

Rule: 7

if the job is writing
or the job is typing
or the job is drawing
then the stimulus_response is documented

Rule: 8

if the job is evaluating
or the job is reasoning
or the job is investigating
then the stimulus_response is analytical

Rule: 9

if the stimulus_situation is 'physical object'
and the stimulus_response is 'hands-on'
and feedback is required
then medium is workshop

Rule: 10

if the stimulus_s*ituation* is symbolic
and the stimulus_r*esponse* is analytical
and feedback is required
then medium is 'lecture tutorial'

Rule: 11

if the stimulus_s*ituation* is visual
and the stimulus_r*esponse* is documented
and feedback is not required
then medium is videocassette

Rule: 12

if the stimulus_s*ituation* is visual
and the stimulus_r*esponse* is oral
and feedback is required
then medium is 'lecture tutorial'

Rule: 13

if the stimulus_situation is verbal
and the stimulus_response is analytical
and feedback is required
then medium is 'lecture – tutorial'

Rule: 14

if the stimulus_situation is verbal
and the stimulus_response is oral
and feedback is required
then medium is 'role-play exercises'

Object	Allowed values	Object	Allowed values
<i>environment</i>	<p><i>papers</i></p> <p><i>manuals</i></p> <p><i>documents</i></p> <p><i>textbooks</i></p> <p><i>pictures</i></p> <p><i>illustrations</i></p> <p><i>photographs</i></p> <p><i>diagrams</i></p> <p><i>machines</i></p> <p><i>buildings</i></p> <p><i>tools</i></p> <p><i>numbers</i></p> <p><i>formulas</i></p> <p><i>computer programs</i></p>	<i>job</i>	<p><i>lecturing</i></p> <p><i>advising</i></p> <p><i>counselling</i></p> <p><i>building</i></p> <p><i>repairing</i></p> <p><i>troubleshooting</i></p> <p><i>writing</i></p> <p><i>typing</i></p> <p><i>drawing</i></p> <p><i>evaluating</i></p> <p><i>reasoning</i></p> <p><i>investigating</i></p>
<i>stimulus_situation</i>	<p><i>verbal</i></p> <p><i>visual</i></p> <p><i>physical object</i></p> <p><i>symbolic</i></p>	<i>stimulus_response</i>	<p><i>oral</i></p> <p><i>hands-on</i></p> <p><i>documented</i></p> <p><i>analytical</i></p>
		<i>feedback</i>	<p><i>required</i></p> <p><i>not required</i></p>

What sort of environment is a trainee dealing with on the job?

MACHINES

Rule: 3

if the environment is machines

or the environment is buildings

or the environment is tools

then the stimulus_situation is 'physical object'

In what way is a trainee expected to act or respond on the job?

REPAIRING

Rule: 6

if the job is building

or the job is repairing

or the job is troubleshooting

then the stimulus_response is 'hands-on'

Is feedback on the trainee's progress required during training?
REQUIRED

Rule: 9

if the stimulus_situation is 'physical object'
and the stimulus_response is 'hands-on'
and feedback is required
then medium is workshop
medium is **workshop**

Inference techniques in Media-Advisor

- The standard inference technique in Leonardo is backward chaining with opportunistic forward chaining.
- Leonardo also enables the user to turn off either backward or forward chaining, and thus allows us to study each inference technique separately.

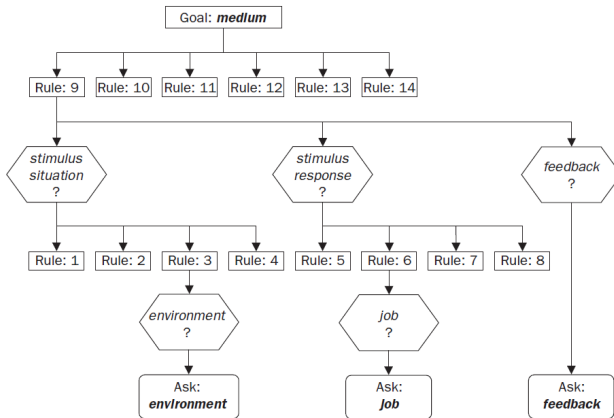


Figure: Tree diagram for the rule-based expert system MEDIA ADVISOR

- 1 CORE CONCEPTS OF EXPERTA
- 2 STEPS TO IMPLEMENT EXPERTA
- 3 DEMONSTRATION OF RULE-BASED (TRAFFIC LIGHT DECISION SYSTEM)
- 4 DEMONSTRATION OF RULE-BASED MEDIA-ADVISOR
- 5 CONFLICT RESOLUTION**
 - conflict resolution methods
- 6 METAKNOWLEDGE
- 7 ASSIGNMENT

Rule 1:

IF the 'traffic light' is green

THEN the action is go

Rule 2:

IF the 'traffic light' is red

THEN the action is stop

Rule 3:

IF the 'traffic light' is red

THEN the action is go

- Remember earlier traffic light rule-based expert system example.
- If the light is Green, and there is a policeman enforces the drivers to stop, or versa visa.
- There is conflict case in this situation, how the expert system should solve this.
- A method for choosing a rule to fire when more than one rule can be fired in a given cycle is called **conflict resolution**.
- If the traffic light is red, which rule should be **executed**?

If the traffic light is red, which rule should be executed?

In forward chaining, both rules would be fired. Rule 2 is fired first as the topmost one, and as a result its THEN part is executed and linguistic object action obtains value stop. However, Rule 3 is also fired because the condition part of this rule matches the fact 'traffic light' is red, which is still in the database. As a consequence, object action takes new value go. This simple example shows that the rule order is vital when the forward chaining inference technique is used.

Are there any other conflict resolution methods?

- Fire the rule with the highest priority.

Goal 1. Prescription is? Prescription

RULE 1 Meningitis Prescription1

(Priority 100)

IF Infection is Meningitis

AND The Patient is a Child

THEN Prescription is Number₁

AND Drug Recommendation is Ampicillin

AND Drug Recommendation is Gentamicin

AND Display Meningitis Prescription1

RULE 2 Meningitis Prescription2

(Priority 90)

IF Infection is Meningitis

AND The Patient is an Adult

THEN Prescription is Number₂

AND Drug Recommendation is Penicillin

AND Display Meningitis Prescription2

- Fire the most specific rule.

Rule 1:

IF the season is autumn

AND the sky is cloudy

AND the forecast is rain

THEN the advice is 'stay home'

Rule 2:

IF the season is autumn

THEN the advice is 'take an umbrella'

- Fire the rule that uses the data most recently entered in the database.

Rule 1:

IF the forecast is rain [08:16 PM 11/25/96]

THEN the advice is 'take an umbrella'

Rule 2:

IF the weather is wet [10:18 AM 11/26/96]

THEN the advice is 'stay home'

Outlines

- 1 CORE CONCEPTS OF EXPERTA
- 2 STEPS TO IMPLEMENT EXPERTA
- 3 DEMONSTRATION OF RULE-BASED (TRAFFIC LIGHT DECISION SYSTEM)
- 4 DEMONSTRATION OF RULE-BASED MEDIA-ADVISOR
- 5 CONFLICT RESOLUTION
 - conflict resolution methods
- 6 METAKNOWLEDGE
- 7 ASSIGNMENT

- As a program grows larger and more complex, it becomes increasingly difficult for the knowledge engineer to manage and oversee rules in the knowledge base.
- To improve the performance of an expert system, we should supply the system with some knowledge about the knowledge it possesses or, in other words, **metaknowledge**.
- Metaknowledge can be simply defined as **knowledge about knowledge**.

Role of Knowledge Engineer

- Transfers domain knowledge from experts to the system.
- Learns to apply problem-specific rules.
- Develops **metaknowledge**: domain-independent knowledge about the system's behavior.

Examples of Metarules

- **Metarule 1:** Expert-supplied rules \searrow Novice-supplied rules.
- **Metarule 2:** Human life-saving rules \searrow Equipment overload rules.

Metarules in Expert Systems

- Some systems have a separate inference engine for metarules.
- Most systems cannot differentiate rules and metarules.
- **Priority:** Metarules are given the highest priority in the knowledge base.
- **Action:** A metarule injects crucial data into the system, altering rule priorities.

Outlines

- 1 CORE CONCEPTS OF EXPERTA
- 2 STEPS TO IMPLEMENT EXPERTA
- 3 DEMONSTRATION OF RULE-BASED (TRAFFIC LIGHT DECISION SYSTEM)
- 4 DEMONSTRATION OF RULE-BASED MEDIA-ADVISOR
- 5 CONFLICT RESOLUTION
 - conflict resolution methods
- 6 METAKNOWLEDGE
- 7 ASSIGNMENT

Conflict Resolution in a Media Advisor Expert System

Question:

- You are tasked with designing a Conflict Resolution Mechanism for a Media Advisor Expert System.

`git@github.com:mtahrawi/Expert-systems.git`

Tasks:

- Define a scenario where conflicting rules might arise.
- Propose a Conflict Resolution Mechanism for the given scenario. Your solution should:
 - 1 Describe the method(s) used to resolve the conflict (e.g., priority, recency, specificity).
 - 2 Include an explanation of why your chosen method is appropriate.
 - 3 Create a sample set of rules and demonstrate how your mechanism resolves conflicts between them.
- Upgrading your mechanism using the attached python work.



THANKS