



| Lecture Notes No. 12 | | | |
|----------------------|--|----------------|-----------------------------|
| Topic: | Case-Based & Model-Based Reasoning | Week No. | 16 |
| Course Code: | CSST101 | Term: | 1 st Semester |
| Course Title: | Advance Knowledge Representation and Reasoning | Academic Year: | 2025-2026 |
| Student Name | | Section | |
| Due date | | Points | |

Learning Outcomes

By the end of this lesson, you should be able to:

1. Explain the concepts of case-based reasoning (CBR) and model-based reasoning (MBR).
2. Retrieve and adapt past cases to solve new problems.
3. Apply model-based reasoning to predict system behavior.
4. Implement simple CBR and MBR tasks in Python.

1. What is Case-Based Reasoning (CBR)?

Case-Based Reasoning solves new problems by **reusing solutions from similar past cases**.

Example:

- Previous case: “Patient with fever and cough → diagnosed with flu”
- New case: “Patient with fever, cough, and sore throat”
- Solution: Adapt previous diagnosis to account for new symptoms

Definition:

CBR involves four main steps:

1. **Retrieve** – Find similar past case(s)
2. **Reuse** – Apply previous solution to the current problem
3. **Revise** – Modify the solution if necessary
4. **Retain** – Store the new case for future use

2. What is Model-Based Reasoning (MBR)?

Model-Based Reasoning uses a **causal or functional model** of the system to understand and predict behavior.

Example:

- Diagnosing why a car won’t start using a mechanical model
- Predict which components could be faulty based on observations



Key Idea:

- CBR relies on **past experiences**
- MBR relies on **domain models and system understanding**

3. Representing CBR in Python

Example: Simple case retrieval and adaptation

```
cases = [  
    {"symptoms": {"fever": True, "cough": True}, "diagnosis": "flu"},  
    {"symptoms": {"fever": True, "rash": True}, "diagnosis": "measles"}  
]  
  
def retrieve_case(new_symptoms):  
    for case in cases:  
        if case["symptoms"] == new_symptoms:  
            return case["diagnosis"]  
    return "No exact match, consider adapting a similar case."  
  
new_case = {"fever": True, "cough": True}  
print(retrieve_case(new_case)) # Output: flu
```

Exercise:

- Add more cases.
- Implement a similarity measure to find the closest match.

4. Representing MBR in Python

Example: Using a simple causal model

```
def diagnose_car(starts, battery_ok, fuel_ok):  
    if not starts:  
        if not battery_ok:  
            return "Battery problem"  
        elif not fuel_ok:  
            return "Fuel problem"  
    else:  
        return "Check engine"  
    return "Car starts normally"  
  
print(diagnose_car(False, True, False)) # Output: Fuel problem
```



Exercise:

- Extend the model with additional components.
- Predict outcomes for different scenarios.

5. Lab Activity (Python)

Goal: Implement CBR and MBR tasks for problem-solving.

Sample Tasks:

- Build a small case library and implement retrieval and adaptation functions.
- Model a system for MBR and simulate predictions.
- Compare results from CBR and MBR approaches.

Challenge:

- Handle conflicting information in cases.
- Implement hybrid reasoning using both CBR and MBR.

6. Applications in AI

| Field | Example of Use |
|-------------------|--|
| Medical Diagnosis | CBR: reuse previous patient cases; MBR: simulate disease progression |
| Fault Diagnosis | MBR: predict system failures using causal models |
| Customer Support | CBR: solve customer issues based on previous solutions |
| Robotics | MBR: predict robot actions using environment models |

7. Reflection and Discussion

1. How does case-based reasoning mimic human problem-solving?
2. When is model-based reasoning more advantageous than CBR?
3. Can you think of real-world scenarios where combining both methods would help?

8. Summary

- Case-Based Reasoning reuses **solutions from past similar cases**.
- Model-Based Reasoning uses **domain models** to predict system behavior.
- Python can implement both CBR and MBR for small-scale problem-solving.
- Combining CBR and MBR allows flexible and intelligent reasoning in complex domains.

Self-Check

1. What are the four steps of case-based reasoning?
2. How does model-based reasoning differ from case-based reasoning?
3. How can Python be used to simulate CBR and MBR?
4. Give an example of a domain where both CBR and MBR could be applied.