Intelligent Decision Support Systems

(Part IX – DATA-DRIVEN MODELS IN DECISION SUPPORT: PREDICTIVE MODELS)

Miquel Sànchez-Marrè

Intelligent Data Science and Artificial Intelligence Research Centre (IDEAI-UPC)
Knowledge Engineering and Machine Learning Group (KEMLG-UPC)
Computer Science Dept.
Universitat Politècnica de Catalunya · BarcelonaTech

miquel@cs.upc.edu
http://www.cs.upc.edu/~miquel

Course 2020/2021

https://kemlg.upc.edu
PART 9 – DATA-DRIVEN MODELS FOR DECISION SUPPORT

Predictive Models

https://kemlg.upc.edu
Predictive Models

https://kemlg.upc.edu
Predictive Models

- Predictive Models
  - (IA) Case-Based Reasoning
  - (Stats) Simple and Multiple Linear Regression, Variance Analysis, Time Series Models.
  - (IA&Stats) Regression Trees, Model Trees
Predictive Models
Case-Based Reasoning (CBR)

Instance-Based Learning (IBL)

https://kemlg.upc.edu
What is CBR? (1)

- A definition:

  “.... transferring knowledge from past problem solving episodes to new problems that share significant aspects with corresponding past experience and using the transferred knowledge to construct solutions to new problems.”

[Carbonell, 1986]
What is CBR? (2)

- **CBR**: A methodology of solving new problems by adapting the solutions of previous similar problems.

  ![Diagram](image)

- It uses **cases** as an episodic memory (**Case Library**).
What is CBR? (3)

- “Similar problems have similar solutions”
CBR vs RBR? (1)

- Why CBR?
  - **Problem**: Most of the time the trouble in building Expert Systems comes from trying to fit *experience* into *rules*.
    
    - Usually it is hard for an expert in a domain the *abstraction* needed to create generic rules from specific past episodes.
CBR vs RBR (2)

- The CBR Solution:
  - to directly use experience (past episodes) in the reasoning (*reason by analogy*).
  - No translation is needed.
CBR Antecedents

● Foundations of CBR
  ■ Our general knowledge about situations is recorded as scripts [Schank & Abelson, 1977]
  ■ Cognitive model is the *Theory of Dynamic Memory* [Schank, 1982]:
    ♦ Indexing is the key to use experience in understanding
    ♦ Remembering, understanding, experiencing, and learning cannot be separated from each other
    ♦ Human memory is dynamic, and change as a result of its experiences

● CBR derives from a view of understanding problem-solving as an explanation process [Riesbeck & Schank, 1989]
Case-Based Reasoning Cycle

Retrieve

new case

retrieved cases

best case

Adapt

adapted solution

Eval

case to store

Learn

evaluated solution (fail/success)

CASE LIBRARY

DOMAIN KNOWLEDGE
Examples
Extracted from [Kolodner, 1993]

- CHEF [Hammond, 1986, 1989], a case-based planner for recipe creation
- CASEY [Koton 1988, 1989], a case-based diagnosis program to diagnose a causal explanation of the patient disorders
- JULIA [Hinrichs 1988-1992], a case-based designer in the domain of meal planning
- HYPO [Ashley, 1990], a case-based interpretive program that works in the domain of law
- PROTOS [Bareiss, 1989], a case-based classification program for audiological disorders
- CLAVIER [Hennessy & Hinkle, 1992], a case-based program for configuration of the layout of composite airplane parts for curing in autoclave
- BATTLE [Goodman, 1989], a case-based program for battle planning
- ARCHIE [Pearce et al., 1992], a case-based design program for architecture
- MEDIATOR [Simpson, 1985], a case-based arbitration program for dispute resolution
Components of a CBR System

- Cases
  - Flat or structured
- Case Library/Case Base
  - Flat memory or hierarchical/network memory
- Retrieval Methods
  - Search or indexing in the Case Library
  - Similarity Assessment
- Adaptation Methods
- Evaluation Methods
- Kind of Learning
Case Representation

- Attribute-value representation: a case is a set of features
  - case identifier
  - derivation of the case
  - description of the problem
  - diagnostic of the problem
  - solution to the problem
  - evaluation of the solution (success/failure)
  - utility measure
  - other relevant information

- Structured representation: a case is a structure relating features and other elements
  - tree or network
Case Retrieval (1)

- Case Retrieval is more difficult than retrieval/query in Databases.
  - DB Recuperation = exact “matching”
  - CBR Retrieval = partial “matching” (similarity)

- Similarity assessment:
  - Computed between case descriptions,
  - Usually, it is an heuristic function or distance,
  - It can be dependent on the domain.

- An example:
  - Case structure = Feature-Value vector
  - Similarity measure \[ \text{similarity}(C_i, C_j) = \sum_{k=1}^{n} w_k \times \text{atr} \_ \text{dist}(C_{ik}, C_{jk}) \]

- Retrieval tries to maximize the similarity between the current case and the case(s) and the retrieved cases.
Case Retrieval (2)

- The efficiency of the retrieval process hardly depends on the **Organization of the Case Library**
- Two main approaches:

  - **Flat memories**:
    - Easy to manage
    - Slow for retrieval
    - Always finds the best

  - **Hierarchical memories**:
    - Hard to manage
    - Fast for retrieval
    - Heuristic search

- The Case Library structure and the Case representation makes easier the relevant case retrieval and its comparison against the current problem.
Adaptation

- When the retrieved case does not perfectly match the new case, then the old solution must be adapted to obtain the new one.

- Strategies:
  - Null adaptation
  - Structural Adaptation
    - Substitution methods
    - Transformation methods
    - Adaptation ad-hoc (special-purpose)
  - Derivational Adaptation

- Adaptation is a highly domain-dependent process.
Evaluation

- Qualify the quality of a solution
- Three basic ways:
  - Testing the proposed solution in the real world
  - Asking to a human expert
  - Executing a simulation model (laboratory, computerized simulation, etc.)
Learning

- Learning by observation (set of initial cases)
- Learning by experience
  - Learning from successful experiences
  - Learning from failed experiences
CBR Applicability

- When a large historical data repository is available
- When experts describe their domain through examples
- When experience is so valuable as the knowledge from textbooks
- When problems are not completely understood (weak domain models, poor domain knowledge)
- There are too many exceptions to general knowledge
- *When cases with similar solutions have similar problem descriptions*
**CBR Applications**

- Failure machine diagnosis
- Computer Network diagnosis
- Medical diagnosis
- Bank Credit analysis
- Geological source prediction
- Battle planners
- Message Classification
- Speech recognition
CBR Advantages

- Fast solution proposal, as it does not start from scratch, using previous experiences
- Easiness to extract expert or domain knowledge to create the case library
- Past failed experiences can be used to prevent making the same mistakes in the future
- Integration of learning skills is simple
  - CBR system improves its performance along time
- Exceptional cases could be easily managed
CBR Shortcomings

- The whole Case Base is not always explored, and thus non-optimal solutions could be found
- A large size of memory could be required
- Global consistency of all the cases could be difficult to maintain
- Adaptation functions must be defined for each domain.
- A CBR system cannot reason about what has never happened
Comparison against other methods

**Rule-Based Reasoning**
- Rules express generic knowledge (*patterns*)
- Rules used in the inferential process *exactly* match with the input problem
- It is difficult to learn new rules and maintain the consistency
  - Static knowledge
  - No learning skills
- It is difficult to acquire the expert knowledge to build the Rule Base
- Performance is constant

**Case-Based Reasoning**
- Cases express specific or episodic knowledge (*constants*)
- Cases used in the inferential process *partially* match with the input problem
- It is easy to learn new cases, storing them in the Case Base
  - Dynamic knowledge
  - Learning skills
- It is relatively easy to acquire the expert knowledge to build the Case Base
- Performance improves along time
Predictive Models

(Stats) Multiple Linear Regression

https://kemlg.upc.edu
Models for the CPU performance data

\[
PRP = -56.1 + 0.049 \text{ MYCT} + 0.015 \text{ MMIN} + 0.006 \text{ MMAX} + 0.630 \text{ CACH} - 0.270 \text{ CHMIN} + 1.46 \text{ CHMAX}
\]
Predictive Models

(AI & Stats) Hybrid Regression Models

https://kemlg.upc.edu
Regression Tree

- Splitting criterion at node $T \equiv$

Attribute Max

$$SDR = sd_T(PrVar) - \sum_{i} \frac{|T_i|}{|T|} * sd_{T_i}(PrVar)$$

- Stopping criterion at leaf $T_l \equiv$

$$\frac{sd_{T_l}(PrVar)}{sd_{T_0}(PrVar)} < p\%, \quad p \approx 5\%$$

Models for the CPU performance data

© Miquel Sànchez i Marrè, KEMLG-IDEAI, 2020
Model Tree

Models for the CPU performance data
Intelligent Data Science and Artificial Intelligence (IDEAI-UPC)

Miquel Sànchez-Marrè
miquel@cs.upc.edu

Knowledge Engineering and Machine Learning Group
UNIVERSITAT POLITÈCNICA DE CATALUNYA

https://kemlg.upc.edu