

8 *Case-based Retrieval*

Literature:

Bergmann, Ralph: Experience Management. Springer-Verlag 2002

Case-Based Reasoning

Assumption: Similar problems have similar solutions

General approach:

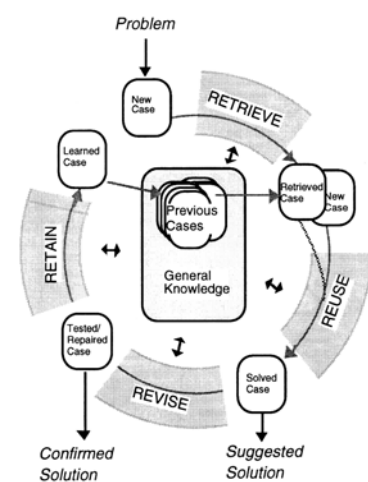
- experiences are stored as cases
- To solve a new problem ...
 - ... similar cases are retrieved
 - ... experiences of the most similar case are reused in the new situation

Humans use Cases for Problem Solving

Examples:

- A medical doctor remembers the case history of another patient
- A lawyer argues with similar original precedence
- An architect studies the construction of existing building
- A work scheduler remembers the construction steps of a similar workpiece (Variantenplanung)
- A mathematician tries to transfers a known proof to a new problem
- A service technician remembers a similar defect at another device

CBR Cycle



- Retrieve ...
 - ◆ most similar case or cases
- Reuse ...
 - ◆ the information and knowledge in that case to solve the problem
- Revise ...
 - ◆ the proposed solution is necessary
- Retain ...
 - ◆ the parts of this experience likely to be useful for future problem solving

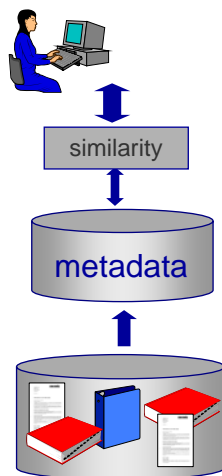
Source: K.-D. Althoff & A. Aamodt: Relating case-based problem solving and learning methods to task and domain characteristics. AI Communications 1996

Case-Based Reasoning for Information Retrieval

- Case-based retrieval can be used for retrieval of
 - ◆ cases (problem-solution pairs) –
 - similarity of problem description with query (=problem)
 - ◆ documents or information –
 - similarity of metadata with query
- Case-based retrieval is useful if
 - ◆ relevant information cannot be specified exactly or
 - ◆ answers do not exactly fit to the query
- Applications examples
 - ◆ Information Retrieval: used cars
 - ◆ Lessons Learned databases: experience management



Example: Information Retrieval in Used Cars Database

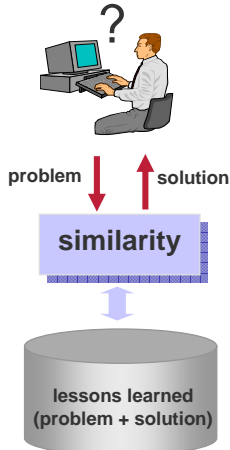


- Scenario: The database contains advertisements of used cars
- Assume you look for the following car:
 - ◆ Audi A4, limousine, 125 PS, colour silver, automatic transmission, 2 years, bis 30'000 Fr.
- Problem: The database does not contain a car with exact this equipment
- Objective: The system should suggest those cars that are most similar to the one I look for, e.g..
 - ◆ Audi A4, limousine, 150 PS, blue, 5 gears, 2 years, 28.000 Fr.
 - ◆ Audi A4, station waggon, 125 PS, silver, automatic transmission, 3 years, 26.000 Fr.
 - ◆ BMW 320, limousine, 138 PS, silver, automatic transmission, 2 years, 29.500 Fr.
 - ◆ Volvo S40, limousine, 125 PS, silver, automatic transmission, 18 months, 29.000 Fr.

What are the similarity criteria for these suggestions?



Lessons Learned Database



- Situation: Hotline for Product NixGeht of Tec Inc.
 - ◆ Complex problems require extensive error diagnostics
- Objective: avoid extensive error diagnostics if problem is already known
 - „Lessons Learned“ database
- But: What if the problem occurs in modified form?
- Example:
 - ◆ Lessons learned database contains an interface product for application in version 2.5 and database version 8.03
 - ◆ Another customer has the same problem with application version 2.7 and database 8.04
- How to find the solution?
- How to transfer the solution?

Example: Diagnosis of Broken Car

CASE 1

Symptoms:

- ◆ problem: driving light not burning
- ◆ type of car: VW Golf
- ◆ year of construction: 2002
- ◆ glass pane: OK
- ◆ light switch: OK

Solution:

- ◆ diagnosis: fuse of driving light broken
- ◆ repair: replace fuse of driving light

CASE 2

Symptoms:

- ◆ problem: driving light not burning
- ◆ type of car: Audi A4
- ◆ year of construction: 2004
- ◆ glass pane: cracked
- ◆ light switch: OK

Solution:

- ◆ diagnosis: bulb of driving light broken
- ◆ repair: replace headlights

New Situation:

Symptoms:

- ◆ problem: break light not burning
- ◆ type of car: VW Passat
- ◆ year of construction: 2003
- ◆ glass pane: OK

Comparison of **Symptoms**:
Which case is most similar?

Diagnosis of Broken Car – Reuse Solution

C
A
S
E
1

- Problem (Symptoms):
- ♦ problem: driving light not burning
 - ♦ ..
- Solution:
- ♦ diagnosis: fuse of **driving** light broken
 - ♦ repair: replace fuse of **driving** light

- Problem (Symptoms):
- ♦ problem: **break** light not burning
 - ♦ type of car: VW Passat
 - ♦ year of construction: 2003
 - ♦ glass pane: OK

Revise !

What is the effect of the difference
between the current situation
and the retrieved case ?

- New solution:
- ♦ Diagnosis: fuse of break light broken
 - ♦ repair: replace fuse of break light

If the solution is correct, store it as a new case in the database.

8.1 Case Representation

Textual Approach

Frequently Asked Question 241
Title: Order numbers of CPUs with which communications is possible.

Question: Which order numbers must the S7-CPU's have to be able to run basic communications with SFCs?

Answer: In order to participate in communications via SFCs without a configured connection table, the module concerned must have the correct order number. The following table illustrates which order number your CPU must have to be able to participate in these S7 homogeneous communications.

Conversational Approach

Case: 241
Title: Printer does not work in the new release.

- Q1:** What kind of problem do you have? Printer Problem
Q1: Does the printer perform a self-test? Yes
Q2: Does the printer work with other software? Yes
Q3: Did you just install the software? Yes
Q4: Did you create a printer definition file? Yes
Q5: What release did you install? 4.2

Problem: Installation procedure overrides printer definition
Action: Reinstall the printer from disk 2.3

Structural Approach


Reference : AD8009
Price : 2.25
Input offset voltage : 2 mV
Input bias current : 50 uA
Output voltage : 1.2 V
Output current drive : 175 mA
Single supply : No
PSPS : 70 dB
Number of devices per package : single
Available Package(s) : SOIC

(Bergmann 2002, p. 54ff)

Structural Case Representation

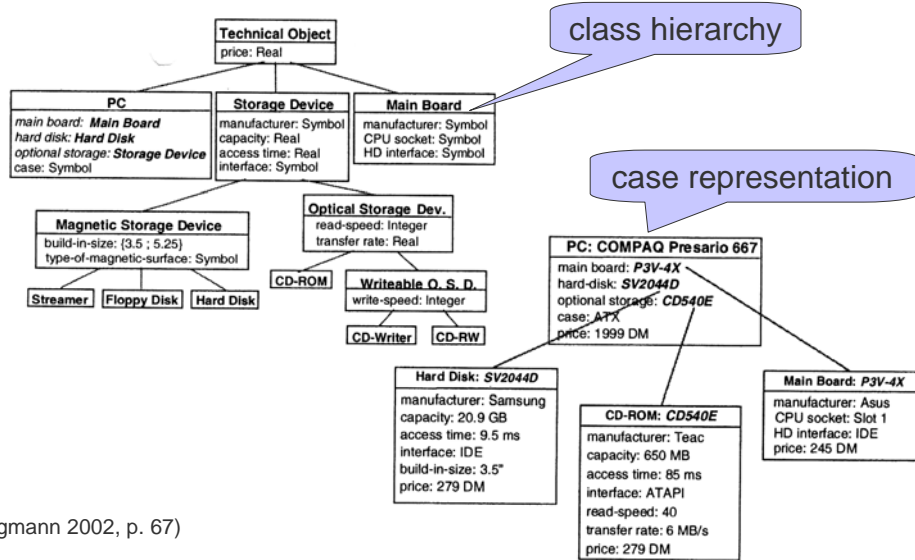
- Attribute Value Representation
- Object-oriented Representation *Our focus*
- Trees and Graphs
- Predicate Logic

Structural Cases: Attribute Value Representation

- The metadata of a case or information is represented with attribute value pairs
Example: Price: 25.000 CHF

 - attribute
 - value
- As in programming language, where types define allowed value ranges for variable, in case representation they define allowed values for attributes. Examples of types are
 - ◆ *numerical types* like integer, real or intervals
 - ◆ *symbol types* defined by
 - enumeration of symbols {red, yellow, green}
 - controlled vocabulary
 - elements of a knowledge structure, e.g. classification scheme
 - ◆ *textual types* such as strings or markups
 - ◆ *special types* for multimedia objects, e.g. a type representing URLs

(Bergmann 2002, p. 62)

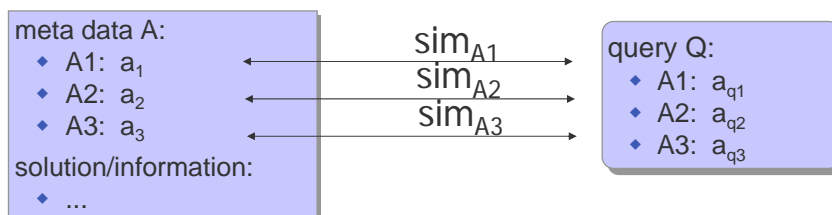
Structural Cases: Object-oriented Representation



(Bergmann 2002, p. 67)

8.2 Similarity Measures for Attribute-Value Pairs

- Cases/metadata are represented by n attributes A_1, \dots, A_n .
 - ◆ each attribute A_i has type T_i



Local similarity: for each attribute a similarity measure is defined

- ◆ $\text{sim}_{A_i}(x_i, y_i): T_i \times T_i \rightarrow [0..1]$
- ◆ local similarity measure depends on the type of the attribute

Global similarity: combining values for local similarity

- ◆ $\text{sim}(A, A') = F(\text{sim}_{A_1}, \text{sim}_{A_2}, \text{sim}_{A_3})$

Similarity Measure

Definition: A *similarity measure* is a function $sim: M \times M \rightarrow [0,1]$

- This definition restricts similarity to a number in the interval $[0,1]$.
 - ◆ It allows to express the most similar (1) and the least similar (0) situation
 - ◆ It also allows to express degrees of similarity: if $sim(x,y) > sim(x,z)$, then x is more similar to y than to z
- Remark: Query answering in SQL can be seen as a special case of a similarity measure where the value range is a set $\{0,1\}$. Two values are either
 - ◆ equal: $sim(x,y) = 1$
 - OR
 - ◆ unequal: $sim(x,y) = 0$

(Bergmann 2002, p. 96)



Usual Properties for Similarity Measures

- A similarity measure is called *reflexive* if
$$sim(x,x) = 1$$
holds for all x ¹⁾
- If additionally $sim(x,y) = 1$ implies that $x = y$, then the similarity measure is called *strong reflexive*
- A similarity measure is called *symmetric* if for all x,y it holds that
$$sim(x,y) = sim(y,x)$$

(Bergmann 2002, p. 101f)

¹⁾ This means that for every value x of M, x is maximally similar to itself



Meanings of Similarity

Similarity ...

... always refers to a specific aspect

- ◆ Two cars are similar if they
 - are of the same brand
 - have similar maximum speed

... is not necessarily transitive

For integers we can say that

- 2 is similar to 4
- 4 is similar to 6
- ...
- 99998 is similar to 100.000

But: Is 2 similar to 100.000?

... is not necessarily symmetric

- ◆ if I look for a limousine, I probably could accept a station wagon
- ◆ if I need a station wagon because of the space, a limousine might not be acceptable for me

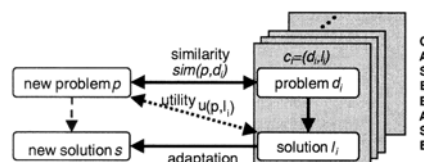


Approximating Utility with Similarity

Assumption

- ◆ Similar problems have similar solutions
- ◆ The solution of a problem is also useful for similar problems

Utility of a case c for a problem p is approximated by the similarity between the problem p and problem d contained in case c



Remark:

- ◆ Utility is an *a posteriori* criteria: It can be assessed **after** after the problem was solved.
- ◆ Similarity is an *a priori* criteria: It must be assessed **before** problem solving.
- ◆ Utility corresponds to relevance in information retrieval

(Bergmann 2002, p. 94)



Local Similarity Measure for Numeric Attributes

- For numeric attributes, similarity is computed as a function of distance d :

$$\text{sim}_A(x,y) = f(|d(x,y)|)$$

- Typical difference functions are the following two:

- ◆ standard linear distance $d(x,y) = x - y$
- ◆ logarithmic distance $d(x,y) = \log(x) - \log(y)$

(logarithmic distance is used if the value range for the attribute spans several orders of magnitude)

- Examples of similarity measure for numeric attributes:

$$\text{sim}_A(x,y) = 1 - \frac{|x-y|}{\max(|x-y|)}$$

$$\text{sim}_A(x,y) = \frac{1}{1 + (|x-y|)}$$

(Bergmann 2002, p. 107)

Symmetric und asymmetric Similarity for Numeric Attributes

$$\text{sim}_{A_i}(x,x) = f(0) = 1$$

Symmetric similarity:

$$\text{sim}_{A_i}(x,y) = f(|\delta(x,y)|)$$

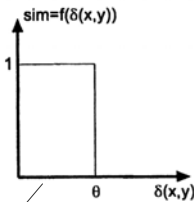
Asymmetric similarity :

$$\text{sim}_{A_i}(x,y) = \begin{cases} f(\delta(x,y)) & : x > y \\ 1 & : x = y \\ g(\delta(y,x)) & : y > x \end{cases}$$

(Bergmann 2002, p. 108)

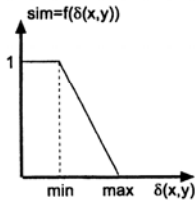
Base Functions for Similarity Measures

Threshold Function



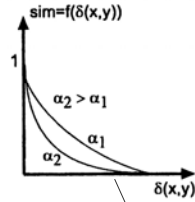
$$f(d) = \begin{cases} 1 & : d < \theta \\ 0 & : d \geq \theta \end{cases}$$

Linear Function



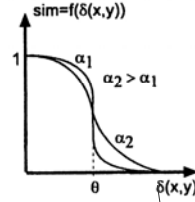
$$f(d) = \begin{cases} 1 & : d < \min \\ \frac{\max-d}{\max-\min} & : \min \leq d \leq \max \\ 0 & : d > \max \end{cases}$$

Exponential Function



$$f(d) = e^{-d \cdot \alpha}$$

Sigmoid Function



$$f(d) = \frac{1}{e^{\frac{d-\theta}{\alpha}} + 1}$$

(Bergmann 2002, p. 108f)



Local Similarity for Ordered Symbols

- For symbolic attributes we can distinguish approaches depending on whether there is an order defined on the symbols or not.
- Example for ordered symbols: qualitative values, e.g. {small, medium, large}
 - ◆ small < medium < large
- With such an order defined, we can determine the similarity by using the ordinal number of the symbols, e.g.
 - ◆ small --> 1
 - ◆ medium --> 2
 - ◆ large --> 3
 and applying similarity measure for numeric attributes



Local Similarity for unordered Symbols

- If there is no obvious ordering on the set of attribute values and no ordering can be defined, we can apply the tabular approach
 - ◆ $sim_A(x,y) = s[x,y]$

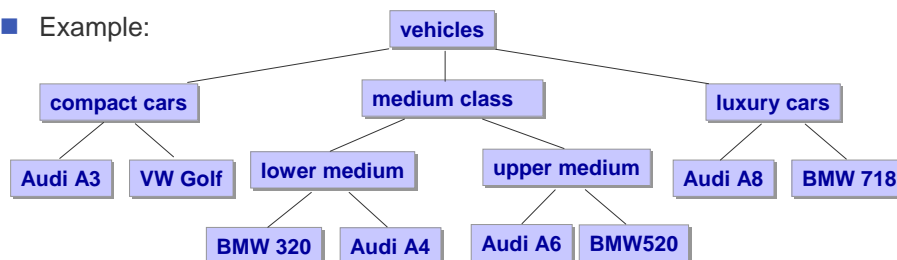
s[x,y]		cases y		
		limousine	convertible	station
query x	limousine	1	0.3	0.7
	convertible	0.4	1	0.2
	station	0.5	0.2	1

similarity of x and y

- Reflexive similarity measure
diagonal values are 1
- Symmetric similarity measure
upper triangular matrix = lower triangular matrix

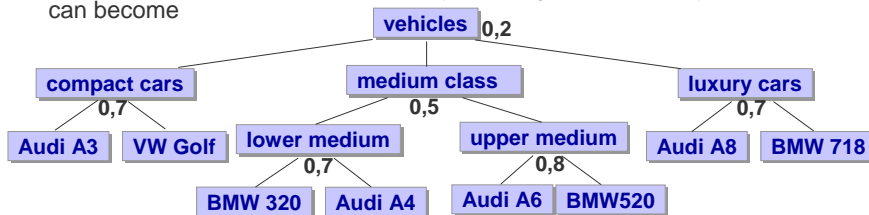
Taxonomically Ordered Symbolic Types

- A special variant of symbolic types are taxonomies. A taxonomy is a tree in which the nodes represent symbolic values
- A taxonomy represents an additional relationship between the symbols
 - ◆ Leaf nodes represent concrete objects of the real world
 - ◆ Inner nodes represent classes of real world objects.
 - ◆ An inner node k stands for the set of real world objects represented by leaf nodes below it
- Example:



Similarity Measure for Taxonomies

- Inner nodes cluster real-world objects that have some properties in common. The deeper we descend in the taxonomy, the more features do the objects have in common
- Similarity measures in a taxonomy
 - ◆ Every inner node K_i is annotated with a similarity value S_i
 - ◆ The deeper the nodes in the hierarchy, the larger the similarity value can become



Similarity of two leaf nodes is the similarity value of the lowest common predecessor

$$\text{sim}(K_1, K_2) = \begin{cases} 1 & : K_1 = K_2 \\ S_{(K_1, K_2)} & : K_1 \neq K_2 \end{cases}$$

- ◆ $\text{sim}(\text{BMW320}, \text{AudiA4}) = 0,7$
- ◆ $\text{sim}(\text{BMW320}, \text{AudiA6}) = 0,5$
- ◆ $\text{sim}(\text{BMW320}, \text{AudiA8}) = 0,2$
(Bergmann 2002, p. 111ff)

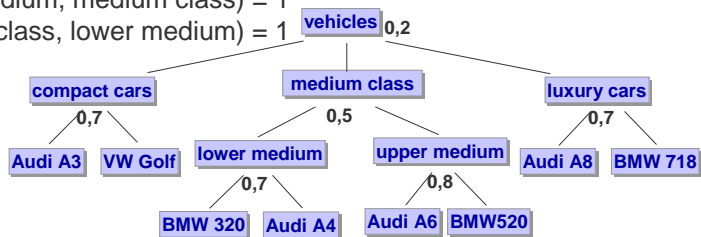
Semantics and Similarity of Inner Nodes

An inner node can have different meanings

- Any value in the query: The inner node in the query stands for any value below this node.
 - ◆ Example: A person is looking for a compact car but does not care whether it is an Audi A3 or a VW Golf
 - ◆ Sample query: „compact car“
- Any value in the Case: The inner value stands for any value below the node
 - ◆ Assume a car dealer specifies that he sells any compact cars
- Uncertainty: The use of an inner node K means that we do not know the exact value for this attribute, but we know that it must be a concrete value below this node
 - ◆ Assume a car dealer specifies that he sells a single compact car without saying whether it is a Audi A3 or a VW Golf

Similarity with inner Nodes of a Taxonomy: Any Value (1/2)

- We are looking for the highest possible similarity $\text{sim}(Q,C)$ ¹⁾
 - ◆ $\text{sim}(\text{AudiA3}, \text{compact car}) = 1$
 - ◆ $\text{sim}(\text{compact car}, \text{AudiA3}) = 1$
 - ◆ $\text{sim}(\text{AudiA4}, \text{medium class}) = 1$
 - ◆ $\text{sim}(\text{medium class}, \text{AudiA4}) = 1$
 - ◆ $\text{sim}(\text{AudiA3}, \text{medium class}) = 0.2$
 - ◆ $\text{sim}(\text{medium class}, \text{AudiA3}) = 0.2$
 - ◆ $\text{sim}(\text{medium class}, \text{compact car}) = 0.2$
 - ◆ $\text{sim}(\text{lower medium}, \text{medium class}) = 1$
 - ◆ $\text{sim}(\text{medium class}, \text{lower medium}) = 1$



¹⁾ The first parameter of $\text{sim}(Q,C)$ is the query, the second parameter is the case description

Similarity with inner Nodes of a Taxonomy: Any Values (2/2)

- Inner node in the query or in the case
 - ◆ All leaf nodes below the inner node have similarity 1
 - ◆ For all other nodes: Take the similarity of the lowest common predecessor.

$$\text{sim}_{A_i}(Q, C) = \max\{\text{sim}(q, c) \mid q \in L_Q, c \in L_C\}$$

$$= \begin{cases} 1 & : C < Q \text{ or } Q < C \\ S_{(Q,C)} & : \text{otherwise} \end{cases}$$

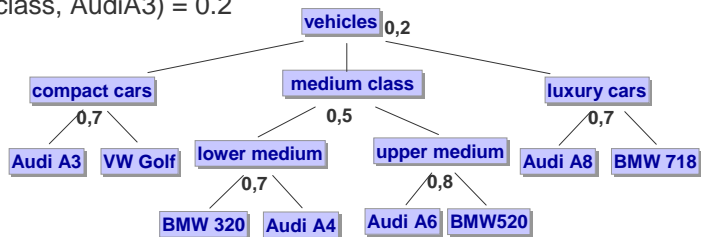
Notation:

Q	Query
C	Case
L_Q, L_C	leaf nodes below Q or C
$C < Q$	C is below Q in the taxonomy (Q is predecessor of C)
$S_{(Q,C)}$	similarity of the lowest common predecessor of Q and C



Similarity with inner Nodes of a Taxonomy: Uncertainty in Query (1/2)

- Optimistic approach computes the upper bound for the similarity $\text{sim}(Q,c)$ ¹⁾
 - ◆ $\text{sim}(\text{compact car}, \text{AudiA3}) = 1$
 - ◆ $\text{sim}(\text{medium class}, \text{AudiA4}) = 1$
 - ◆ $\text{sim}(\text{medium class}, \text{AudiA3}) = 0.2$
- Pessimistic approach computes the lower bound for the similarity $\text{sim}(Q,c)$ ¹⁾
 - ◆ $\text{sim}(\text{compact car}, \text{AudiA3}) = 0.7$
 - ◆ $\text{sim}(\text{medium class}, \text{AudiA4}) = 0.5$
 - ◆ $\text{sim}(\text{medium class}, \text{AudiA3}) = 0.2$



¹⁾ The first parameter of $\text{sim}(Q,C)$ is the query, the second parameter is the case description

Similarity with inner Nodes of a Taxonomy: Uncertainty in Query (2/2)

- Optimistic approach: Upper bound

$$\text{sim}_{A_i}(Q, c) = \max\{\text{sim}(q, c) | q \in L_Q\} = \begin{cases} 1 & : c < Q \\ S_{(Q,c)} & : \text{otherwise} \end{cases}$$

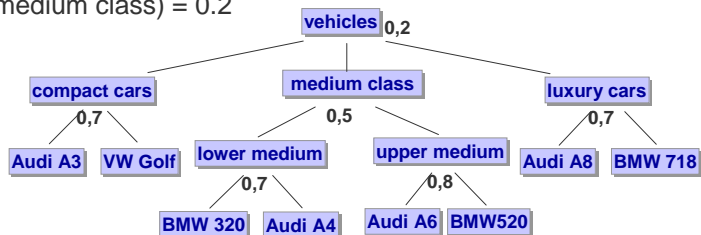
- Pessimistic approach: Lower bound

$$\text{sim}_{A_i}(Q, c) = \min\{\text{sim}(q, c) | q \in L_Q\} = S_{(Q,c)}$$



Similarity with inner Nodes of a Taxonomy: Uncertainty in Case (1/2)

- Optimistic approach computes the upper bound for the similarity $\text{sim}(q,C)$ ¹⁾
 - ◆ $\text{sim}(\text{AudiA3}, \text{compact car}) = 1$
 - ◆ $\text{sim}(\text{AudiA4}, \text{medium class}) = 1$
 - ◆ $\text{sim}(\text{AudiA3}, \text{medium class}) = 0.2$
- Pessimistic approach computes the lower bound for the similarity $\text{sim}(q,C)$ ¹⁾
 - ◆ $\text{sim}(\text{AudiA3}, \text{compact car}) = 0.7$
 - ◆ $\text{sim}(\text{AudiA4}, \text{medium class}) = 0.5$
 - ◆ $\text{sim}(\text{AudiA3}, \text{medium class}) = 0.2$



¹⁾ The first parameter of $\text{sim}(Q,C)$ is the query, the second parameter is the case description

Prof. Dr. Knut Hinkelmann

Case-based Retrieval

30

Similarity with inner Nodes of a Taxonomy: Uncertainty in Case (2/2)

- Optimistic approach: Upper bound

$$\text{sim}_{A_i}(q, C) = \max\{\text{sim}(q, c) | c \in L_C\} = \begin{cases} 1 & : q < C \\ S_{(q,C)} & : \text{otherwise} \end{cases}$$

- Pessimistic approach: Lower bound

$$\text{sim}_{A_i}(q, C) = \min\{\text{sim}(q, c) | c \in L_C\} = S_{(q,C)}$$



Prof. Dr. Knut Hinkelmann

Case-based Retrieval

(Bergmann 2002, p. 118)

31

Multiple Attribute Values

- Attributes can contain multiple values
 - ◆ $A_{\text{query}} = \{a_1, \dots, a_n\}$
 - ◆ $A_{\text{case}} = \{b_1, \dots, b_m\}$
- Similarity measure for sets:
 - ◆ Compute all pairs of similarity measures $sim_A(a_i, b_j)$
 - ◆ Aggregate the local similarity

$$sim_A(A_{\text{query}}, A_{\text{case}}) = MF(sim_A(a_1, b_1), \dots, sim_A(a_1, b_m), \dots, sim_A(a_n, b_1), \dots, sim_A(a_n, b_m))$$
- There are various possible approaches for the aggregate function MF, e.g.
 - ◆ minimum
 - ◆ maximum
 - ◆ average



Derived Attributes

- Occasionally, the attributes themselves are not significant for similarity measurement
- A derived attribute is an attribute, whose value is computed from other attributes
- Example: credits

derived
Attribut

Attribute	case 1	case 2	case 3	query
income:	1000	2000	5000	2000
expenses:	1500	5000	4500	1500
creditworthy:	no	no	yes	?
difference	-500	-3000	500	500

The derived attribute is used for similarity measurement



Unknown Attribute Values

- It often occurs that attribute values are not known (NULL):
- Strategies to deal with unknown values
 - ◆ **optimistic strategie:** Assume that unknown values are most similar: $\text{sim}(\text{NULL},x) = 1$
 - ◆ **pessimistic strategie:** Assume that unknown values are most similar: $\text{sim}(\text{NULL},x) = 0$.
 - ◆ **strategy of expected value:** Use an expected value, e.g. based on probability or average
 - ◆ ignore the attributes



Global Similarity

- Global similarity measures are defined by applying an aggregation function $F : [0..1]^n \rightarrow [0..1]$ to the local similarity values.
 - ◆ Input: Local similarity measures $\text{sim}_{A_i}(x_i, y_i)$ for each attribute A_i
 - ◆ Global similarity:
$$\text{sim}(x,y) = F(\text{sim}_{A_1}(x_1, y_1), \dots, \text{sim}_{A_n}(x_n, y_n))$$
- Possible properties for F
 - F is monotone in each argument
 - $F(0, \dots, 0) = 0$
 - $F(1, \dots, 1) = 1$



Basic Aggregation Functions

■ Weighted Average: $F(s_1, \dots, s_n) = \sum_{i=1}^n w_i \cdot s_i$ with $\sum_{i=1}^n w_i = 1$

■ Generalized weighted average: $F(s_1, \dots, s_n) = \sqrt[\alpha]{\sum_{i=1}^n w_i \cdot s_i^\alpha}$ with $\alpha \in \mathbb{R}^+$ und $\sum_{i=1}^n w_i = 1$

■ Maximum: $F(s_1, \dots, s_n) = \max_{i=1}^n (w_i \cdot s_i)$

■ Minimum: $F(s_1, \dots, s_n) = \min_{i=1}^n (w_i \cdot s_i)$

(Bergmann 2002, p. 120f)

