TDT4173 - Machine Learning and Case-Based Reasoning

Assignment 2

Iver Jordal

1. Theory

Case-Based Reasoning

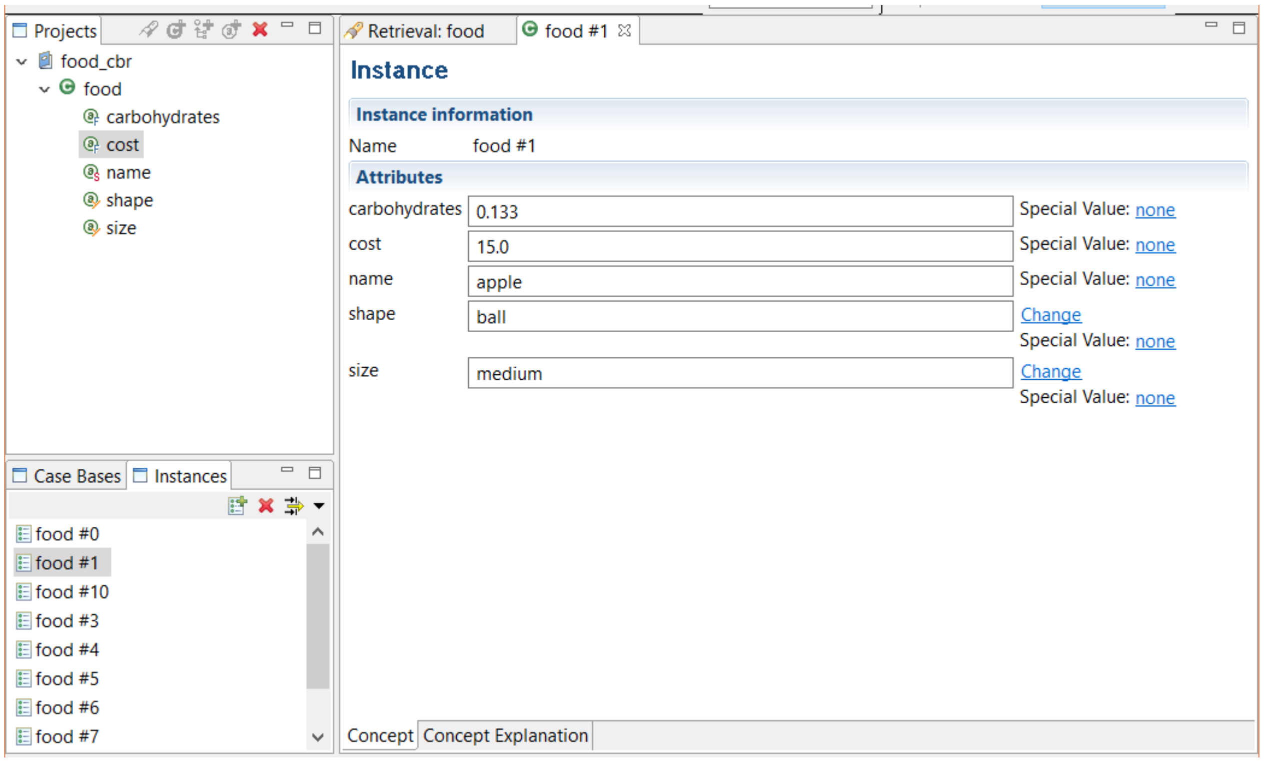
1. Case-Based Reasoning is based on the assumption that similar problems have similar solutions. When solving a new (unseen) problem, CBR methods look for similar problems that were solved in the past, and uses those experiences to propose a solution to the new problem. How are CBR methods different from other machine learning approaches: *"CBR (…) delays (implicit) generalization of its cases until testing time – a strategy of lazy generalization"* –Wikipedia. Other machine learning approaches tend to use eager generalization, i.e. before testing time.
2. CBR is inspired by everyday human problem solving. Humans can remember how they solved a certain problem and apply that knowledge to new problems that are similar. CBR methods also have this concept. They have memory that can store cases that contain descriptions of problems and solutions and mappings from problems to solutions. This memory is called a case base. Also the process of problem solving in CBR has roots in cognitive science. Case-Based problem solving consists of four steps: Retrieving, Reusing, Revising and Retaining.

   "Case-based Reasoning is (...) reasoning by remembering."
   -Leake, 1996

3. Surface similarity: Two objects can *look like* each other, for example a bike might resemble a pair of glasses. However, these aspects are not so relevant when looking up memories to solve similar problems. For example, knowing how to use a pair of glasses does not help much when trying to use a bike. Structural similarity is more about the semantics of a pair of objects. For example a pair of glasses and a bike have totally different structures although they have some similar shapes. Another example: A red car and a yellow car are pretty much structurally similar. They might not look very similar on the surface, but that is not so important when it comes to problem-solving. If you know how to repair a red car, you can pretty much apply that knowledge to yellow cars as well.

2. Practical

Case Modelling



Instance

Instance information

Name	food #1

Attributes

carbohydrates	0.133	Special Value: none
cost	15.0	Special Value: none
name	apple	Special Value: none
shape	ball	Change Special Value: none
size	medium	Change Special Value: none

Figure 1: Screenshot of one of the food instances

Case Retrieval

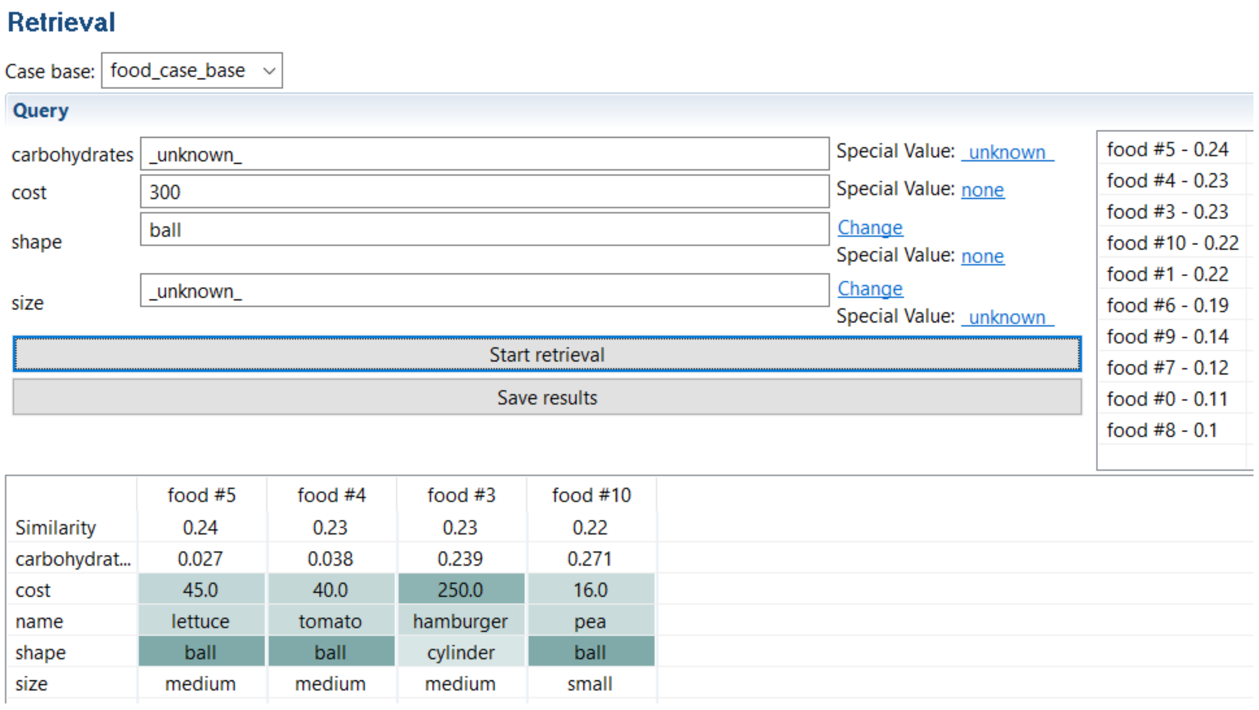
2.

Retrieval queries:

1. Something that has little carbohydrates
   - Carbohydrates: 0.01
2. Something cheap with lots of carbohydrates
   - Cost: 10
   - Carbohydrates: 0.4
3. Something ball-shaped and expensive
   - Cost: 300
   - Shape: ball
4. The food that is the most similar to apple
   - Carbohydrates: 0.133
   - Cost: 15.0
   - Size: medium
   - Shape: ball
5. Something small and expensive
   - Cost: 300
   - Size: small

Retrieval #3 is interesting. Both lettuce and tomato match the shape feature of the query, but lettuce is more expensive, so it comes first. Third comes the hamburger. Although hamburgers are cylinder-shaped, they are somewhat similar to ball-shaped objects. In the similarity function for shapes I've entered 0.3 for similarity between ball and cylinder. Because the hamburger matches the cost well and the shape is *somewhat* similar, it shows up as the third most similar instance.

Furthermore, the similarity scores here are all low. This is because I only specified two attributes in my query. One other factor is that I use weighted sum in the food similarity measure with weight = 0.5 for cost and weight = 0.3 for shape.



carbohydrates	_unknown_	Special Value: _unknown_
cost	300	Special Value: none
shape	ball	Change Special Value: none
size	_unknown_	Change Special Value: _unknown_

food #5 - 0.24
food #4 - 0.23
food #3 - 0.23
food #10 - 0.22
food #1 - 0.22
food #6 - 0.19
food #9 - 0.14
food #7 - 0.12
food #0 - 0.11
food #8 - 0.1

	food #5	food #4	food #3	food #10
Similarity	0.24	0.23	0.23	0.22
carbohydrat...	0.027	0.038	0.239	0.271
cost	45.0	40.0	250.0	16.0
name	lettuce	tomato	hamburger	pea
shape	ball	ball	cylinder	ball
size	medium	medium	medium	small

Figure 2: Screenshot of retrieval #3

3.

In my first retrieval attempts I got unexpected results: all instances got the same similarity and hence the ranking did not match my expectations. I later found out that this was due to a bug in myCBR (at least that's how I perceived it), and I was able to find a workaround: delete the modified default similarity functions and create new ones with other names. After the workaround I did not get any strange or unexpected results during retrieval. Nevertheless, strange/unexpected results can occur for example if the data contains errors/inaccuracies/noise, the model is not accurate enough or the similarity functions are not well tuned to match expectations and domain knowledge.

4.

I'm not sure what "a full CBR cycle for one of my queries" refers to, so I'll assume that I'm supposed to explain the math behind the similary measures and do the calculations by hand.

I'll choose the first query: Carbohydrates: 0.01

This retrieval problem is finding food instances that have carbohydrates values that are close to 0.01.

The similarity function for carbohydrates is $1 - abs(q - x_i)$ where q is the carbohydrates value specified in the query and x is the carbohydrates value of instance i. Furthermore, the food similarity is using weighted sum. The way that works is you sum all similarity terms that are multiplied with their respective weights, and then you divide all that by the sum of weights.

If “full CBR cycle” refers to the what’s going on in Figure 3: I’ve already covered the retrieve part. When it comes to the reuse part; This is about mapping the existing problem(s) (in the case base) to the new problem. I’d just pick the highest ranked result in the retrieval. The revise step is all about tuning the solution after it has been tested in the real world. When the solution is good, it should be retained, i.e. stored in memory (aka the case base).

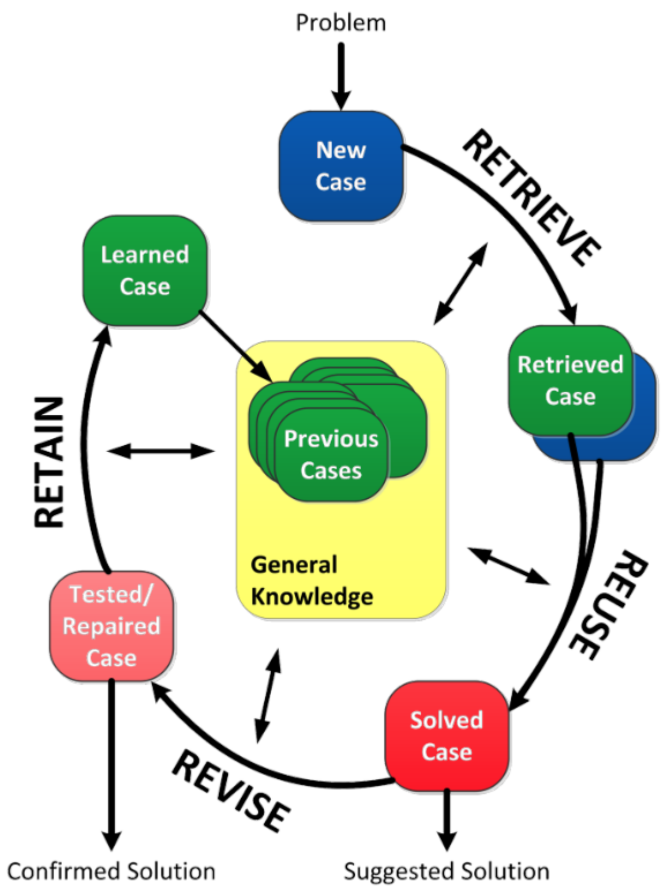


Figure 3: Case-Based Reasoning Cycle from a slide in lecture 4