The Virtue of Reward:

Performance, Reinforcement and Discovery in Case-Based Reasoning

Derek Bridge
University College Cork
Ireland

International Conference on Case-Based Reasoning 2005
Invited talk

Overview

- Motivation
- Reinforcement Learning
- Case-Based Classifier Systems
- Preliminary Results
- Concluding Remarks

Overview

- ➢ Motivation
- Reinforcement Learning
- Case-Based Classifier Systems
- Preliminary Results
- Concluding Remarks

Reasoning and Acting over Time

- Single-step problems, solved repeatedly
  - e.g. spam classification
- Multi-step (episodic) problems
  - e.g. dialogue management
- Continuous problem-solving
  - e.g. factory process control

Reasoning and Acting over Time

- Early experiences may be
  - unrepresentative
  - suboptimal
  - rendered incorrect by change (“concept drift”)
- Agents must be highly adaptive
- CBR may be well-suited
  - robust, incremental lazy learners

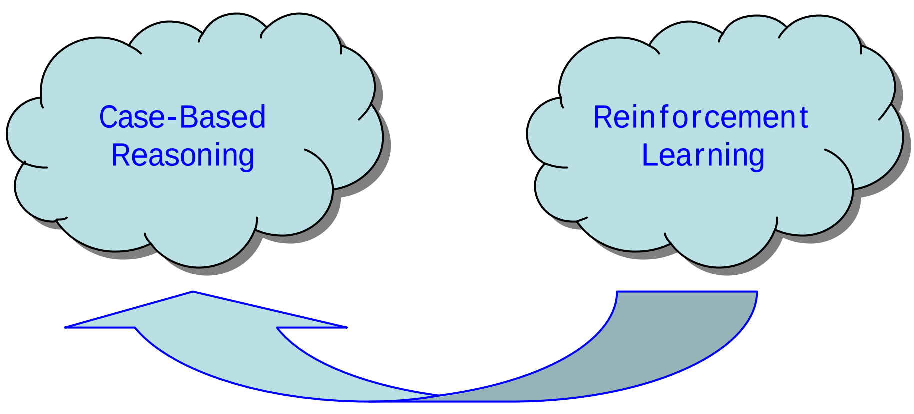
Too much CBR research assumes...

- ...an up-front training set...
- ...of correctly labelled examples (supervised learning)...
- ...for a classification task...
- ...in a stationary environment.

Notwithstanding...

- Incremental learning (e.g. Aha et al. 1991); active learning/selective sampling (e.g. Wiratunga et al. 2003)
- Case base maintenance, esp. noise elimination (e.g. Wilson & Martinez 2000)
- Optimisation problems (Miyashita & Sycara 1995); control problems (e.g. Kopeikina et al. 1988); plus planning, design, etc.
- Concept drift in spam classification (e.g. Cunningham et al. 2003); cache-based statistical models of language (e.g. Kuhn & De Mori 1990)

CBR for Agents that Reason and Act over Time



Reinforcement Learning

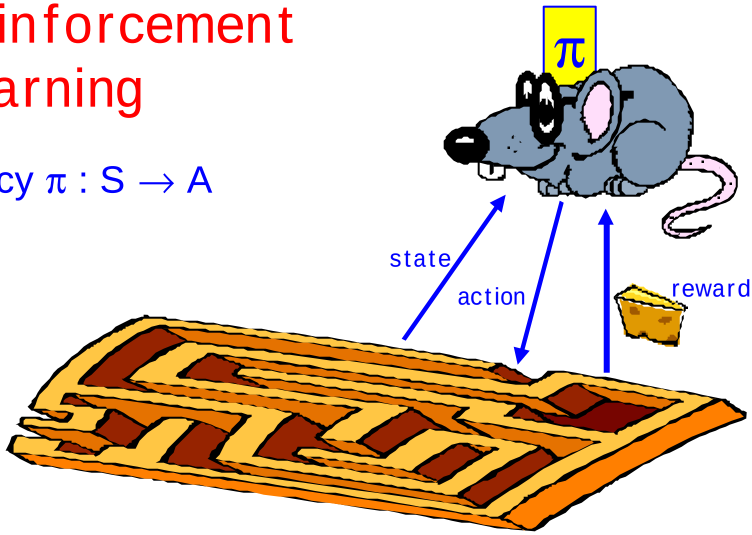
- The agent interacts with its environment to achieve a goal
- It receives reward (possibly delayed reward) for its actions
  - it is not told what actions to take
- Trial-and-error search
  - neither exploitation nor exploration can be pursued exclusively without failing at the task
- Life-long learning
  - on-going exploration

Overview

- ✓ Motivation
- ➢ Reinforcement Learning
- Case-Based Classifier Systems
- Preliminary Results
- Concluding Remarks

Reinforcement Learning

Policy $\pi : S \rightarrow A$



$$s_0 \xrightarrow[r_0]{a_0} s_1 \xrightarrow[r_1]{a_1} s_2 \xrightarrow[r_2]{a_2} \cdots$$

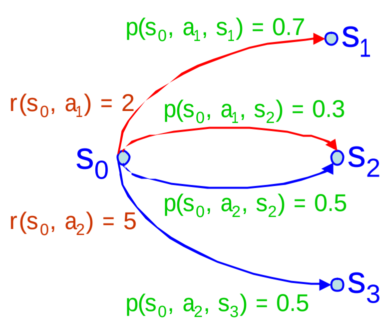
State value function, V

State, *s*	V(*s*)
s_0	...
s_1	10
s_2	15
s_3	6

V(*s*) predicts the future total reward we can obtain by entering state *s*

π can exploit V greedily, i.e. in *s*, choose action *a* for which the following is largest:

$$r(s,a) + \sum_{s' \in S} p(s,a,s') \cdot V(s')$$



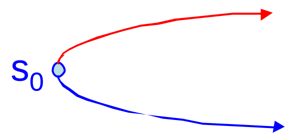
Choosing a_1: $2 + 0.7 \times 10 + 0.3 \times 15 = 13.5$
Choosing a_2: $5 + 0.5 \times 15 + 0.5 \times 6 = 15.5$

Action value function, Q

State, s	Action, a	$Q(s, a)$
s_0	a_1	13.5
s_0	a_2	15.5
s_1	a_1	...
s_1	a_2	...

$Q(s, a)$ predicts the future total reward we can obtain by executing a in s

π can exploit Q greedily, i.e. in s, choose action a for which $Q(s, a)$ is largest



Q Learning

For each (s, a), initialise $Q(s, a)$ arbitrarily

Observe current state, s

Do until reach goal state

Select action a by exploiting Q e-greedily,
i.e. with probability e, choose a randomly;
else choose the a for which $Q(s, a)$ is largest

Execute a, entering state s' and receiving
immediate reward r

Update the table entry for $Q(s, a)$

s ¬ s'

Watkins 1989

Q Learning

For each (s, a), initialise Q(s, a) arbitrarily

Observe current state, s

Do until reach goal state

Select action a by exploiting Q ε-greedily

immediate reward r

Update the table entry for Q(s, a)

$s \leftarrow s'$

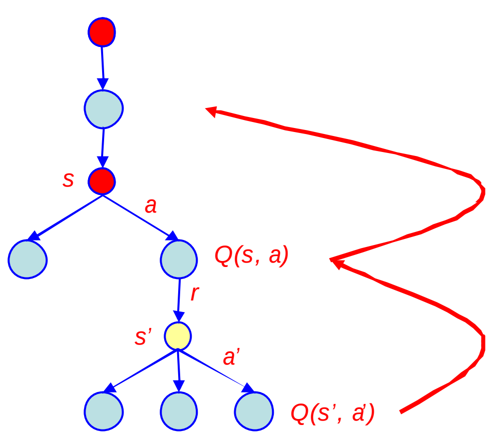
Exploration versus exploitation

One-step temporal difference update rule, TD(0)

$$Q(s,a) \leftarrow Q(s,a) + \alpha\,(r + \gamma \max_{a'} Q(s',a') - Q(s,a))$$

Watkins 1989

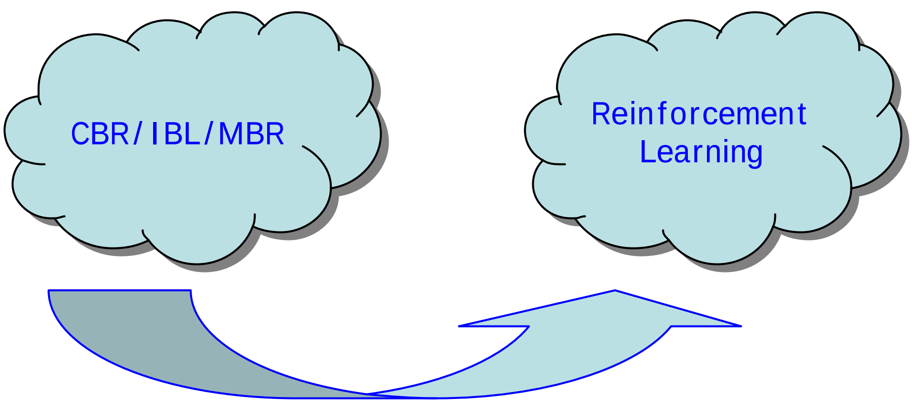
Backup Diagram for Q Learning



Function Approximation

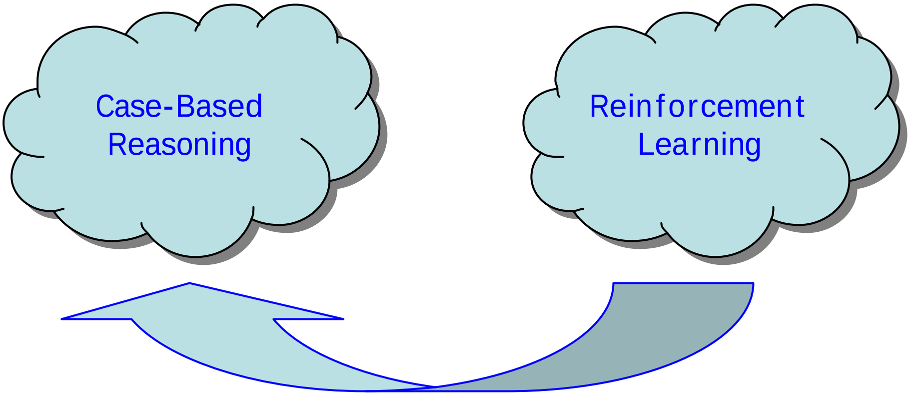
- Q can be represented by a table only if the number of states & actions is small
- Besides, this makes poor use of experience
- Hence, we use function approximation, e.g.
  - neural nets
  - weighted linear functions
  - case-based/instance-based/memory-based representations

CBR/IBL/MBR for RL



Driessens & Ramon 2003; Forbes & Andre 2002; Gabel & Riedmiller 2005; McCallum 1995; Santamaría et al. 1998; Smart & Kaelbling 2000; ...

RL's Influence on CBR



Ram & Santamaría 1997; Zeng & Sycara 1995

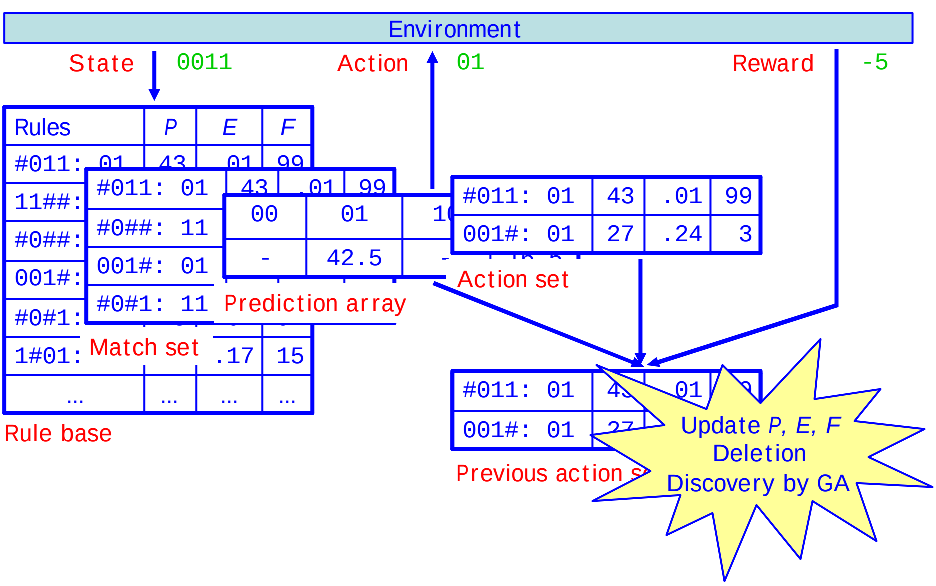
Overview

- ✓ Motivation
- ✓ Reinforcement Learning
- ➢ Case-Based Classifier Systems
- Preliminary Results
- Concluding Remarks

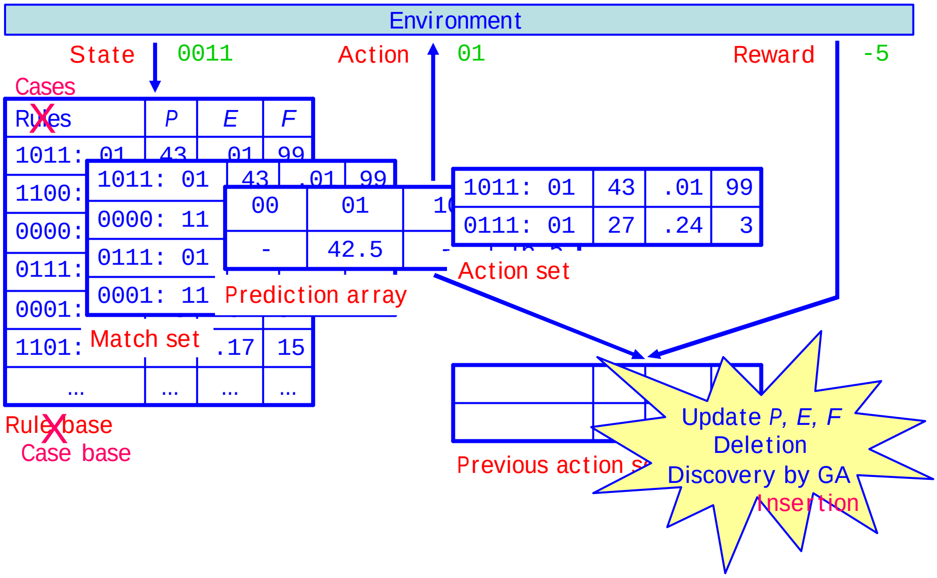
Classifier Systems

- John Holland 
  - Classifier systems are rule-based systems with components for performance, reinforcement and discovery [Holland 1986]
  - They influenced the development of RL and GAs
- Stewart Wilson 
  - ZCS simplifies Holland's classifier systems [Wilson 1994]
  - XCS extends ZCS and uses accuracy-based fitness [Wilson 1995]
  - Under simplifying assumptions, XCS implements Q Learning [Dorigo & Bersini 1994]

XCS

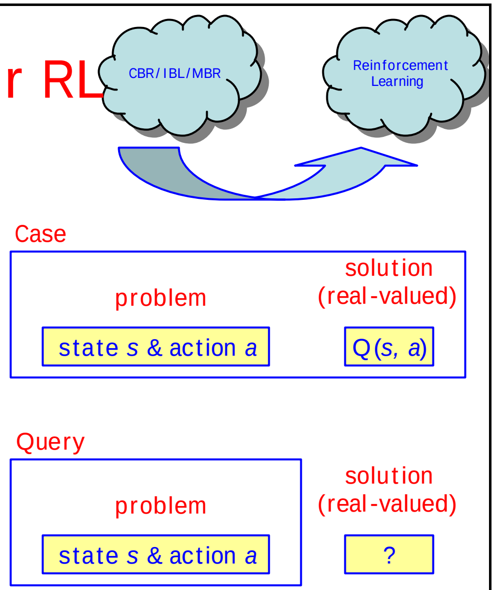


Case-Based XCS

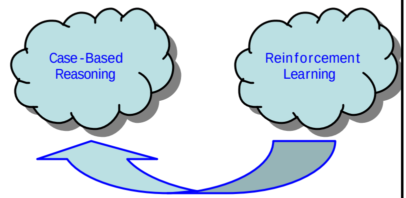


CBR/IBL/MBR for RL

- Conventionally, the case has two parts
  - *problem description*, representing (*s, a*)
  - *solution*, representing Q(*s, a*)
- Hence, the task is *regression*, i.e. given a new (*s, a*), predict Q(*s, a*) (real-valued)



Case-Based XCS



- Case has three parts
  - *problem description*, representing *s*
  - *solution*, representing *a*
  - *outcome*, representing Q(*s*, *a*)

Case

problem	solution	outcome (real-valued)
state *s*	action *a*	Q(*s*, *a*)

- Given new *s*, predict *a*, guided by case outcomes as well as similarities

Query

problem	solution
state *s*	?

Case Outcomes

- In CBR research, storing outcomes is not common but neither is it new, e.g.
  - cases have three parts in [Kolodner 1993]
  - IB3's classification records [Aha et al. 1991]

- They
  - influence retrieval and reuse
  - are updated in cases, based on performance
  - guide maintenance and discovery

Outcomes in Case-Based XCS

- Each case outcome is a record of
  - *experience*:
    how many times it appeared in an action set
  - *prediction of future reward, P:*
    this is its estimate of Q(*s*, *a*)
  - *prediction error, E:*
    average error in *P*
  - *fitness, F:*
    inversely related to *E*

Retrieval and Reuse

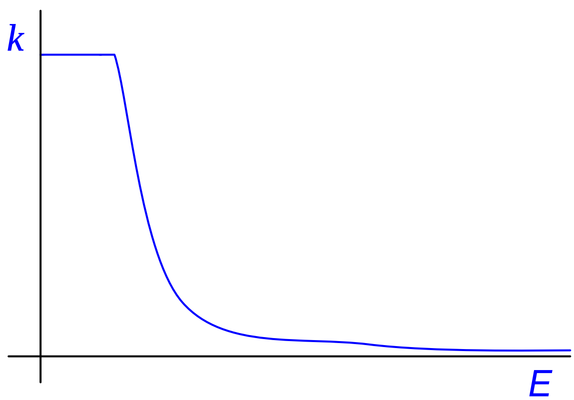
- The Match Set contains the *k*-nearest neighbours, but similarity is weighted by fitness

- From the Prediction Array, we choose the action with the highest predicted total future reward, but the cases' predictions are weighted by similarity and fitness

Reinforcement

- On receipt of reward *r*, for each case in the Previous Action Set
  - *P* is updated by the TD(0) rule
  - *E* is moved towards the difference between the case's previous value of *P* and its new value of *P*
  - *F* is computed from accuracy k, which is based on error *E*

Fitness *F* and Accuracy k



Fitness *F* is accuracy k relative to the total accuracies of the Previous Action Set

Deletion

- ‘Random’ deletion
  - probability inversely related to fitness

- Or case c_i might be deleted if there is another case c_j such that
  - c_j has sufficient experience
  - c_j has sufficient fitness (accuracy)
  - c_j subsumes c_i i.e.
    - sim(c_i, c_j) < ? (or could use a competence model)
    - c_j's action = c_i's action

Discovery by GA

- Steady-state reproduction, not generational
- The GA runs in a niche (an action set), not panmictically
- It runs only if time since last GA for these cases exceeds a threshold
- From the action set, two parents are selected; two offspring are created by crossover and mutation
- They are not retained if subsumed by their parents
- If retained, deletion may take place

Overview

- ✓ Motivation
- ✓ Reinforcement Learning
- ✓ Case-Based Classifier Systems
- ➢ Preliminary Results
- Concluding Remarks

Spam Classification

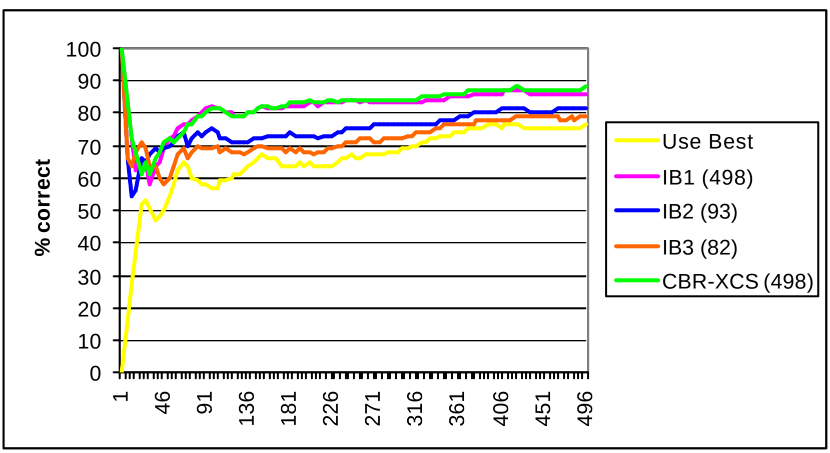
- Emails from my mailbox, stripped of attachments
  - 498 of them, approx. 75% spam
  - highly personal definition of spam
  - highly noisy
  - processed in chronological order
- Textual similarity based on a *text compression ratio*
- $k = 1; e = 0$
- No GA

Spam Classification

- Rewards
  - correct: 1
  - spam as ham: -100
  - ham as spam: -1000

- Other ways of reflecting this asymmetry
  - skewing the voting [Delany et al. 2005]
  - loss functions, e.g. [Wilke & Bergmann 1996]

Has Spam had its Chips?



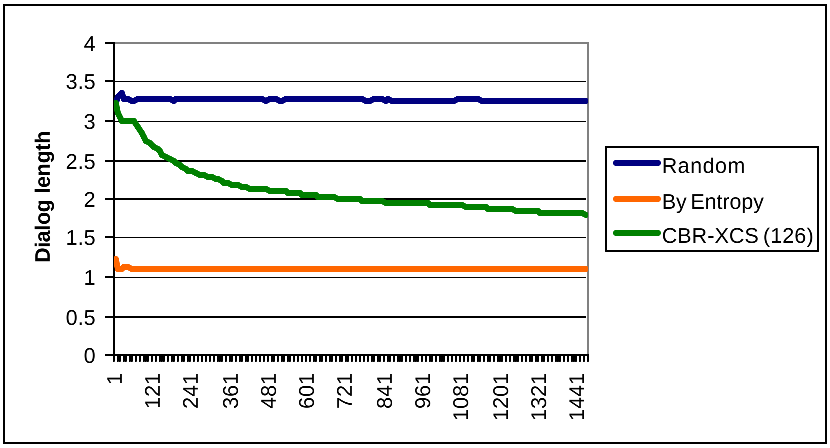
Recommender System Dialogs

- 1470 holidays; 8 descriptive attributes
- Leave-one-in experiments
  - each holiday in turn is the target holiday
  - questions are asked until retrieval set contains ≤ 5 holidays or no questions remain
  - simulated user answers a question with the value from the target holiday
  - 25-fold cross-validation (different orderings)

Users Who Always Answer

- Best policy is to choose the remaining question that has highest entropy
- State, *s*, records the entropies for each question
- $k = 4$; e starts at 1 and, after ~150 steps, decays exponentially
- Delayed reward = - (numQuestionsAsked^3)
- Multi-step backup
- No GA

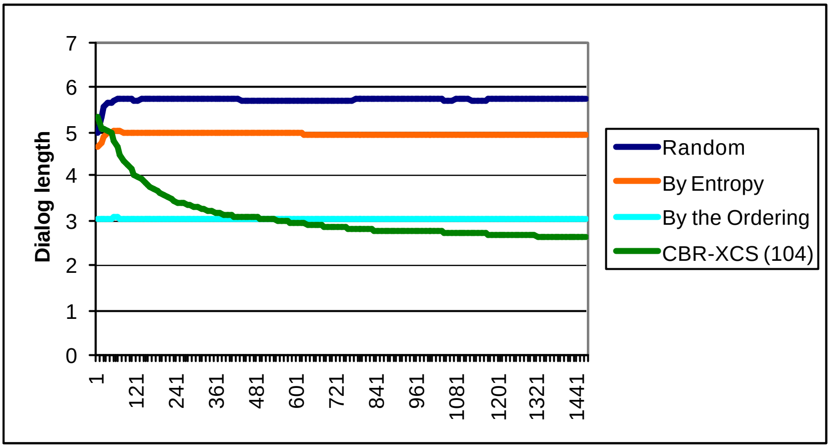
Does the learned policy minimise dialog length?



Users Who Don't Always Answer

- Schmitt 2002:
  - an entropy-like policy (*simVar*)
  - but also customer-adaptive (a Bayesian net predicts reaction to future questions based on reactions to previous ones)
- Suppose users feel there is a 'natural' question order
  - if the actual question order matches the natural order, users will always answer
  - if actual question order doesn't match the natural order, with non-zero probability users may not answer
- A trade-off
  - learning the natural order
    - to maximise chance of getting an answer
  - learning to ask highest entropy questions
    - to maximise chance of reducing size of retrieval set, if given an answer

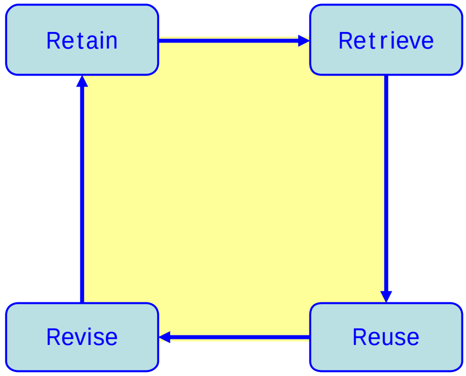
Does the learned policy find a good trade-off?



Overview

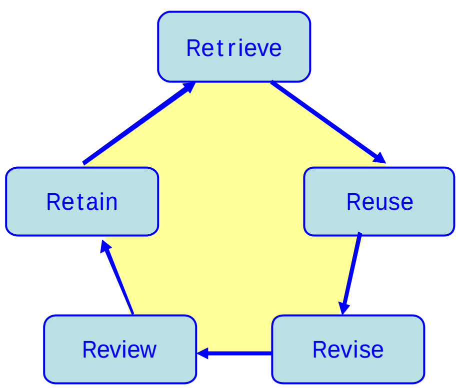
- ✓Motivation
- ✓Reinforcement Learning
- ✓Case-Based Classifier Systems
- ✓Preliminary Results
- ➢Concluding Remarks

Aamodt's & Plaza's 4 REs



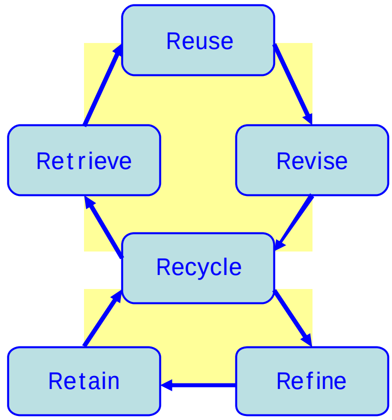
Aamodt & Plaza 1994

Aha's 5 REs



Aha 1998

Göker's & Roth-Berghofer's 6 REs



Göker & Roth-Bergofer 1999

Revealing

(for the first time)

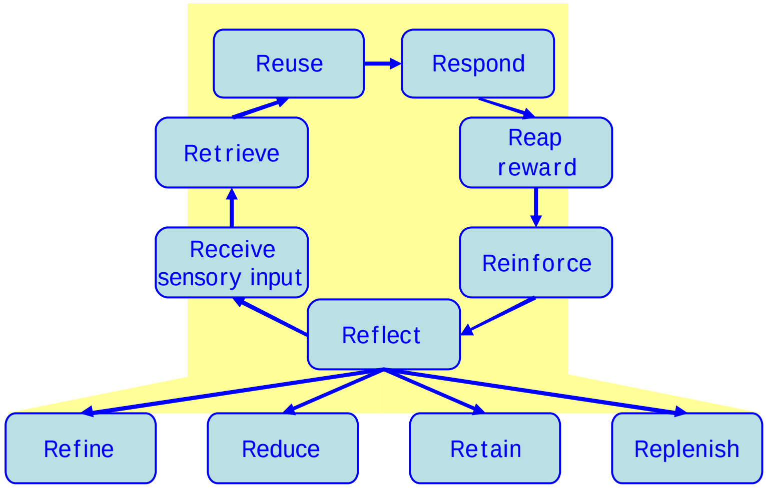
Derek Bridge's

11 REs

for Case-Based Agents

that Reason and Act over Time

Bridge's 11 REs



Thank you for your attention

I hope it was rewarding

References

- Aamodt, A. & Plaza, E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches, *AI Communications*, vol.7(1), pp.39-59, 1994
- Aha, D.W.: The Omnipresence of Case-Based Reasoning in Science and Application, *Knowledge-Based Systems*, vol.11(5-6), pp.261-273, 1998
- Aha, D.W., Kibler, D & Albert, M.K.: Instance-Based Learning Algorithms, *Machine Learning*, vol.6, pp.37-66, 1991
- Cunningham, P., Nowlan, N., Delany, S.J. & Haahr, M.: A Case-Based Approach to Spam Filtering that can Track Concept Drift, *Long-Lived CBR Systems Workshop, 5th ICCBR*, pp.115-123, 2003
- Delany, S.J., Cunningham, P., Tsymbal, A. & Coyle, L.: A Case-Based Technique for Tracking Concept Drift in Spam Filtering, *Knowledge-Based Systems*, vol.18(4-5), pp.187-195, 2005
- Dorigo, M. & Bersini, H.: A Comparison of Q-Learning and Classifier Systems, *3rd International Conference on Simulation of Adaptive Behavior*, pp.248-255, 1994
- Driessens, K. & Ramon, J.: Relational Instance Based Regression for Relational Reinforcement Learning, *20th ICML*, pp.123-130, 2003
- Forbes, J. & Andre, D.: Representations for Learning Control Policies, ICML Workshop on Development of Representations, pp.7-14, 2002
- Gabel, T. & Riedmiller, M.: CBR for State Value Function Approximation in Reinforcement Learning, *6th ICCBR*, 2005
- Göker, M. & Roth-Berghofer, T.: Development and Utilization of a Case-Based Help-Desk Support System in a Corporate Environment, *3rd ICCBR*, pp.132-146, 1999
- Holland, J.H.: Escaping Brittleness: The Possibilities of General-Purpose Learning Algorithms Applied to Parallel Rule-Based Systems, in R.S. Michalski et al., *Machine Learning: An Artificial Intelligence Approach, Volume II*, pp.593-623, Morgan Kaufmann, 1986
- Kolodner, J.: *Case-Based Reasoning*, Morgan Kaufmann, 1993
- Kopeikina, L., Brandau, R. & Lemmon, A.: Case-Based Reasoning for Continuous Control, *DARPA Workshop on Case-Based Reasoning*, pp.233-249, 1988

References continued

- Kuhn, R. & De Mori, R.: A Cache-Based Natural Language Model for Speech Reproduction, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.12(6), pp.570-583, 1990
- McCallum, R.A.: Instance-Based Utile Distinctions for Reinforcement Learning with Hidden State, *12th ICML*, pp.387-395, 1995
- Miyashita, K. & Sycara, K.: CABINS: A Framework of Knowledge Acquisition and Iterative Revision for Schedule Improvement and Reactive Repair, *Artificial Intelligence Journal*, vol.76(1-2), pp.377-426, 1995
- Ram, A. & Santamaría, J.C.: Continuous Case-Based Reasoning. *Artificial Intelligence*, vol.90(1-2), pp.25-77, 1997
- Santamaría, J.C., Sutton, R.S. & Ram, A.: Experiments with ReinforcementLearning in Problems with Continuous State and Action Spaces, *Adaptive Behavior*, vol.6(2), pp.163-218, 1998
- Schmitt, S.: *simVar:* A Similarity-Influenced Question Selection Criterion for e-Sales Dialogs, *Artificial Intelligence Review*, vol.18(3-4), pp.195-221, 2002
- Smart, W.D. & Kaelbling, L.P.: Practical Reinforcement Learning in Continuous Spaces, *17th ICML*, pp.903-910, 2000
- Watkins, C.J.C.H.: *Learning from Delayed Rewards*, Ph.D. thesis, University of Cambridge, 1989
- Wilke, W. & Bergmann, R.: Considering Decision Cost During Learning of Feature Weights, *3rd EWCBR*, pp.460-472, 1996
- Wilson, D.R. & Martinez, T.R.: Reduction Techniques for Instance-Based Learning Algorithms, *Machine Learning*, vol.38, pp.257-286, 2000
- Wilson, S..W.: ZCS: A Zeroth Level Classifier System, *Evolutionary Computation*, vol.2(1), pp.1-18, 1994
- Wilson, S.W.: Classifier Fitness Based on Accuracy, *Evolutionary Computation*, vol.3(2), pp.149-175, 1995
- Wiratunga, N., Craw, S. & Massie, S.: Index Driven Selective Sampling for CBR, *5th ICCBR*, pp.637-651, 2003
- Zeng, D. & Sycara, K.: Using Case-Based Reasoning as a Reinforcement Learning Framework for Optimization with Changing Criteria, 7th International Conference on Tools with Artificial Intelligence, pp.56-62, 1995