CS.760

Case-Based Reasoning 7

Dr. Ian Watson

 www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz

Contents

- Competence Models (cont.)
  - The retrieval problem
  - Case footprints
  - Footprint-based retrieval
  - An experiment
- Case-Based Maintenance
- Project ideas

 www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz

Retrieval

- the retrieval problem
  - Efficiency vs. Accuracy
- the bottom line ...
  - We'd prefer not to examine every case during retrieval but don't want to miss the best case!

 www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz

Retrieval

- case-base reduction
  - nearest-neighbor editing techniques {CNN, RNN,...}
  - reduced search costs
  - optimality/quality sacrifices
- footprint-based retrieval
  - competence-directed search
  - search reduction without quality sacrifices

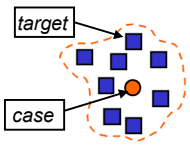
Competence

- what is competence?
  - the set of target problems that a case can solve
  - ... and vice versa
  - the set of cases that can solve a target problem
- the representativeness assumption
  - treat the case-base as a representative sample of the target problem space

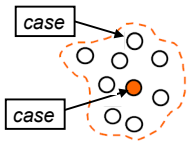
coverage sets

The ***coverage set*** of a ***case*** *is the set of* ***target*** *problems that it can solve.*

coverageset(●) = { ■ 's }



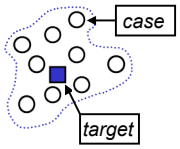
... or, by the representativeness assumption ...



The ***coverage set*** of a ***case*** *is the set of* ***cases*** *that it can solve.*

coverageset(●) = { ○ 's }

reachability sets



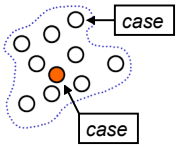
The ***reachability set*** of a ***target*** *is the set of* ***cases*** *that can solve it.*

reachabilityset(■) = { ○'s }

... or, by the representativeness assumption ...

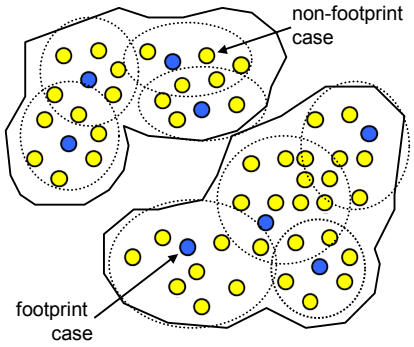
The ***reachability set*** of a ***case*** *is the set of* ***cases*** *that can solve it.*

reachabilityset(●) = { ○'s }



 www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz

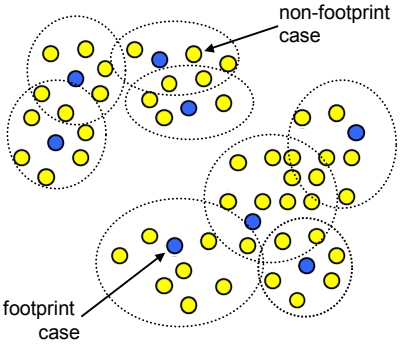
coverage & footprints



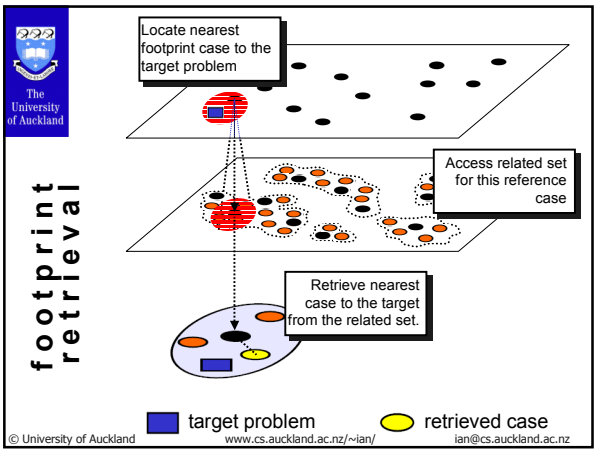
 www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz

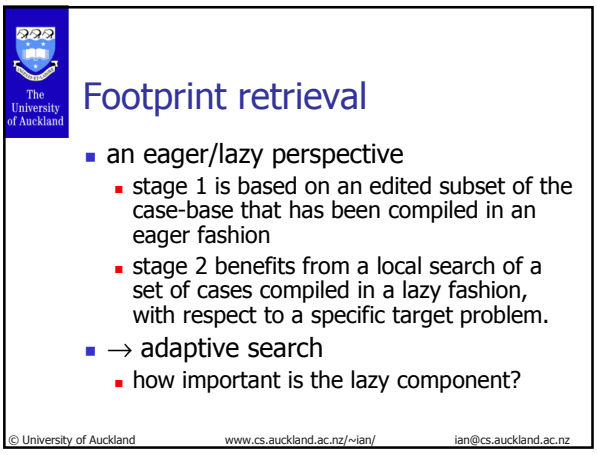
coverage & footprints

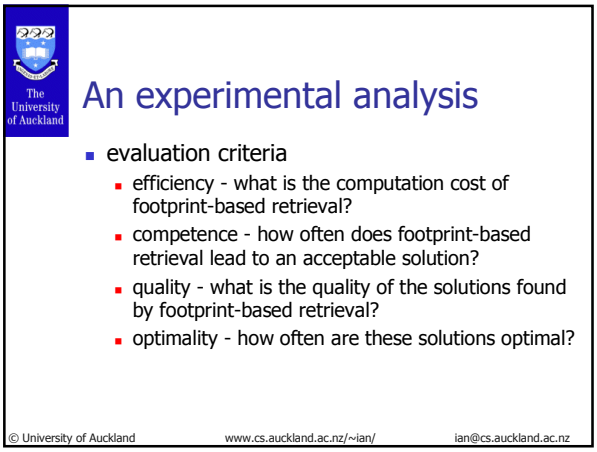
4 footprint cases



 www.cs.auckland.ac.nz/~ian/ ian@cs.auckland.ac.nz


The University of Auckland
footprint retrieval
Locate nearest footprint case to the target problem
Access related set for this reference case
Retrieve nearest case to the target from the related set.
target problem
retrieved case
© University of Auckland
www.cs.auckland.ac.nz/~ian/
ian@cs.auckland.ac.nz


The University of Auckland
Footprint retrieval
an eager/lazy perspective
stage 1 is based on an edited subset of the case-base that has been compiled in an eager fashion
stage 2 benefits from a local search of a set of cases compiled in a lazy fashion, with respect to a specific target problem.
→ adaptive search
how important is the lazy component?
© University of Auckland
www.cs.auckland.ac.nz/~ian/
ian@cs.auckland.ac.nz


The University of Auckland
An experimental analysis
evaluation criteria
efficiency - what is the computation cost of footprint-based retrieval?
competence - how often does footprint-based retrieval lead to an acceptable solution?
quality - what is the quality of the solutions found by footprint-based retrieval?
optimality - how often are these solutions optimal?
© University of Auckland
www.cs.auckland.ac.nz/~ian/
ian@cs.auckland.ac.nz

Datasets

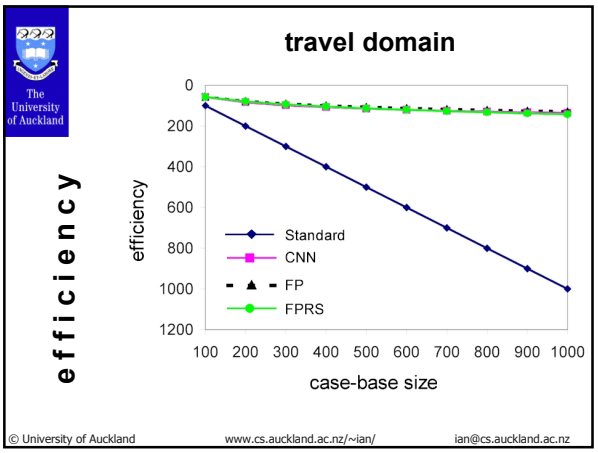
- travel domain (AI-CBR)
  - 1400 case travel data-set
  - 100 sets of 1000 cases and 400 targets
  - each set is used to produce incrementally larger case-bases from 100 to 1000 cases in steps of 100

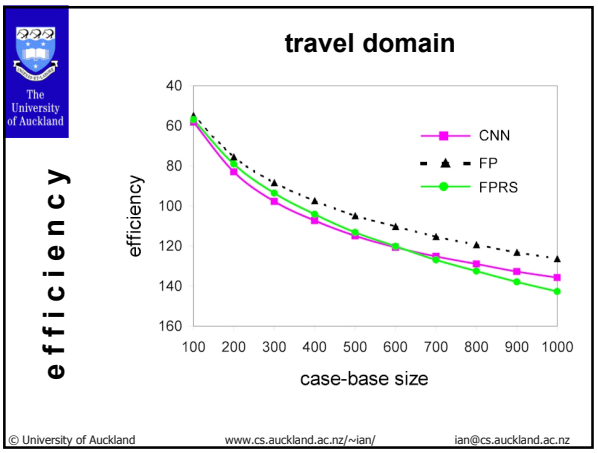
Algorithms compared

- traditional
  - Standard (brute force nearest neighbour)
  - CNN (search CNN-edited case-base)
- footprint-based
  - FP (search footprint set only - stage 1)
  - FPRS (full footprint-based approach)

Efficiency

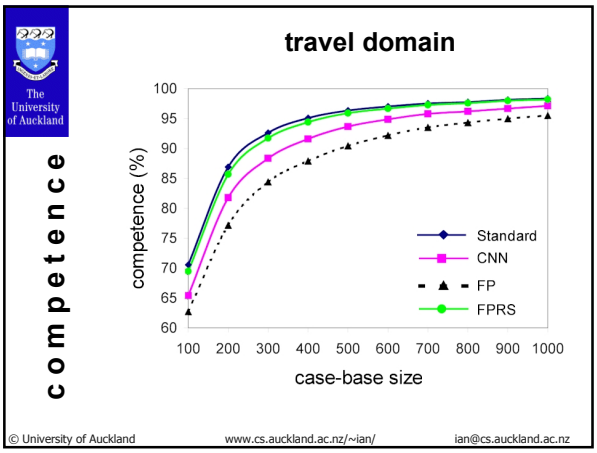
- objective
  - efficiency - number of cases examined during a retrieval
- method
  - for each algorithm and the test case-base we measure the mean number of cases examined during retrieval for a given case-base size





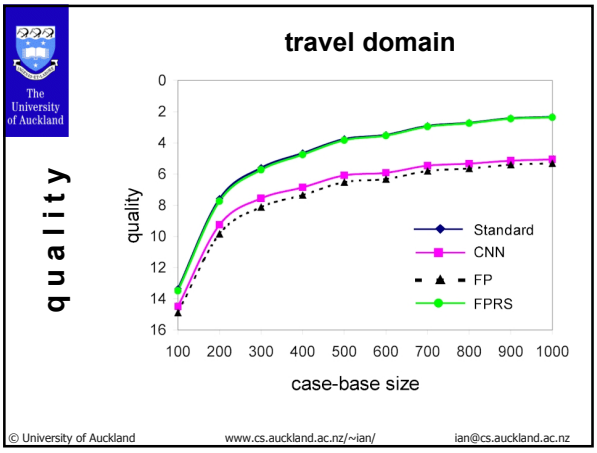
Competence

- objective
  - efficiency vs. competence trade-off?
  - competence - percentage of retrieved cases capable of solving the target
- method
  - for each algorithm and the test case-base we measure the mean percentage of correct retrievals per case-base size



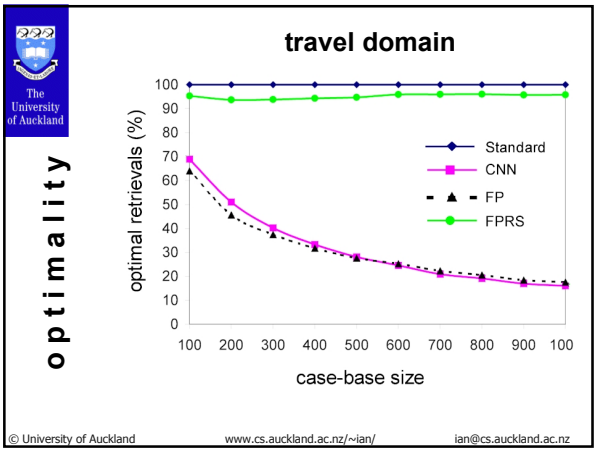
Quality

- objective
  - quality ∝ similarity between retrieved case and target problem
- method
  - for each algorithm and the test case-bases we measure the mean similarity between target problem and retrieved case for each case-base size



Optimality

- objective
  - optimality - percentage of retrievals that return the closest case
- method
  - for each algorithm and the test case-bases we measure the mean optimality during retrieval for a given case-base size



Discussion

- central contribution
  - a novel retrieval technique informed by a explicit model of case competence
- results
  - the competence, quality and optimality characteristics of brute-force retrieval
  - the efficiency benefits of an edited case-base search such as CNN

Characterising Maintenance

- *Categorizing Case-Base Maintenance: Dimensions and Directions* by David B. Leake and David C. Wilson. Advances in Case-Based Reasoning: Proceedings of EWCBR-98, Springer-Verlag, Berlin.

Case-Base Maintenance

- Definition:
  - Case-base maintenance implements policies for revising the organization or contents (representation, domain content, accounting information, or implementation) of the case-base in order to facilitate future reasoning for a particular set of performance objectives (Leake & Wilson, 98)

Case-Base Maintenance (CBM)

- Performace objectives should include
  - Retrieval accuracy or precision
  - Efficiency
  - And usability metrics
- Performance objectives may change over time

CBM Policies

- A CBM Policy determines
  - When a CBR system is maintained
  - How it is maintained
- CBM policies describe
  - What data is gathered
  - How maintenance activities are triggered
  - The types of maintenance operations available
  - How selected operations are executed

CBM Policies

- Data Collection
  - gathers information about individual cases, about the case base in part or as a whole, and/or about the overall processing behaviour of the CBR system
  - Data collection about individual cases might record the number of times a case has been successfully used or the number of times it has failed
  - Data collection about the case base as a whole could involve, for example, monitoring the size of the case base

CBM Policies

- Type of data collected:
  - None – no data recorded, decisions are made independent of the present or past state of the case-base
  - non-introspective

CBM Policies

- Type of data collected:
  - Introspective
    - Synchronic – a singe snapshot in time – eg should the new case be added based upon the current state of the case-base
    - Diachronic – creates a sequence over time enabling trends to be studied – eg recording retrieval time to identify the onset of the utility problem

CBM Policies

- Timing
  - Periodic – done at regular periods w.r.t the CBR-cycle
    - Continous – every time the CBR-cycle is completed
  - Conditional – upon some event – a retrieval failure or when the case base exceeds a set size
  - Ad hoc – just done when ever (eg monthly) or triggered by some event external to the CBR system itself (eg end of a sales quarter or a SMEs decision)

CBM Policies

- Integration
  - On-line – during the reasoning process (as a background activity)
  - Off-line – during a break in reasoning (eg at night or between query sessions)

CBM Policies

- Data Collection results in ***Triggering*** causing maintenance to occur or not
- Conditional Triggering may be in response to a well defined event – eg a retrieval failure
  - Space-based – w.r.t. the case-base size
  - Time-based – w.r.t. retrieval time
  - Result-based – w.r.t. some result (eg retrieval failure)

CBM Policies

- Scope ofMaintenance
  - Broad – applies to the entire case-base or majority of cases
  - Narrow – applies to an individual case or competence group

CBM Policies

- Operation types:
  - Target types:
    - Indexing structures
    - Domain contents
    - Maintenance policies
  - Revision levels:
    - Implementational level (eg algorithm code)
    - Representational level (eg feature names)
    - Knowledge level (eg adding removing cases)

CBM Policies

- Execution timing
  - None – eg maintenance is flagged for a person to attend to
  - Periodic – perfomed on a regular basis
  - Conditional – held until a trigerring event (eg when enough changes have accumulated)

CBM Policies

- Execution integration
  - Off-line
  - On-line
- Execution scope
  - Broad – changes to a single case
  - Narrow – global changes

CBR project ideas

- Create an interesting CBR system to do *X*
- Ensemble retrieval – create a CBR system that combines several ML retrieval algorithms to improve retrieval accuracy (or efficiency) – would require comparative evaluations
- CBR for email filtering – either as an anti-SPAM devise or to answer student/client emails
- CBR for game playing – investigate the use of CBR within gaming