Case-Based Reasoning 6
Dr. Ian Watson

Contents
- Testing & evaluation
  - Leave one out
  - Leave one in
  - Global system metrics
- Case competence
  - The Competence Issue
  - A Competence Model for CBR
  - Case Discovery – competence holes
  - Discussion

Testing
- Testing an ML algorithm is easy (boring but easy)
  - Obtain a data set
  - Divide into training & test data
  - Train your classifier
  - Run the classifier on the test set
- Not so straightforward with CBR
Testing

What are we testing?
- The retrieval (classification) accuracy?
  - Then ML methodology is appropriate
- The adaptation accuracy?
  - Validate the accuracy of the generated solution
- The combined accuracy (retrieval & adaptation)
- What about efficiency, speed?
- Also a case-base changes with time....

The performance of a CBR system is the product of a combination of processes
To test one in isolation may give spurious results
Difficult to scientifically test all together
But there are some useful simple tests

Leave one out testing
- For $i = 1$ to $n$ (where $n =$ No. cases)
- Remove case, and use as a query case
Useful for finding outlying cases
Useful for finding dense areas of the case-base
Testing

- Leave one in testing
  - For $i = 1$ to $n$ (where $n = \text{No. cases}$)
  - Do NOT Remove case, but use as a query case
  - query-case should exactly match case,
  - Also useful for finding outlying cases
  - Also useful for finding dense areas of the case-base

Testing

- Global system tests
  - Considers accuracy as well as performance issues
  - Since a case-base changes with time
  - Create a reference set of cases
  - Log performance data using this reference set over time
  - Useful way of monitoring relative performance

System Performance

- Ratings
  - retrieval speed
  - retrieval precision
  - adaptation accuracy
  - users' perception
Testing

- No point in testing if you know you have a "bad" case-base
- The best retrieval and adaptation algorithms will not work well on a "bad" case base
- But what is a bad or a good case-base?

Characterizing a case-base

- Motherhood statements.....
  - The case-base should be "representative"
  - The cases should be "well" distributed
  - Cases should be useful
- Doesn't really help us much
- Hence case competence models
  - University College Dublin (Barry Smyth)

What is performance

- Performance = Competence + Efficiency
- In pure CBR
  - Cases contribute to both competence and efficiency
Recent Developments in CBR

- Real-World Applications
  - Large Scale Case-Bases
  - On-Line Learning
- Emerging Issues
  - The Utility Problem
    - More cases decreases the utility of individual cases
  - Case-Base Maintenance & Case Quality Issues
  - Authoring & Learning Support

Open Questions

- Quality Issues
  - Good vs Bad Cases / Useful vs Redundant Cases
- Controlling Case-Base Growth
  - Building & Maintaining Quality Case-Bases
- Authoring Tools
  - Case-Based Visualization
  - Authoring Guidance
  - Case Discovery – competence holes

Case Competence: The Basics

- Case Coverage & Case-Base Coverage

Target Problem Space, T

Uncovered Problem

Covered Problems

Case Coverage
Mapping Case Competence

- The Top-Level
  - Problem Space
  - Cases & Target Problems
- Case Coverage
  - Competence vs Efficiency
  - The Utility Problem

Mapping Case Competence

- Competence Groups
  - Independent regions of related competence
  - The fundamental unit of competence

Mapping Case Competence

- Case Competence
  - Competence categories
- Footprint Cases
  - Cases that provide equivalent coverage to the group as a whole
- Non-Footprint Cases
  - Redundant?
Mapping Case Competence

- Competence Holes
  - Uncovered regions and problems

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

\[
\text{coverageset} = \{s\}
\]

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

... or, by the representativeness assumption ... (ie we assume the case-base is representative)

Case-Competence: The Basics

- An Ideal Measure of Case Coverage
  - For a case-base CB and a target problem set T
  - \(\text{Coverage}(c) = \{t \in T : \text{Solves}(c, t)\}\)

- A Practical Measure of Case Coverage
  - CB is a representative sample of T
  - \(\text{Coverage}(c) = \{c' \in CB : \text{Solves}(c, c')\}\)
Case Competence: The Basics

- Approximating Coverage Sets

\[ \text{CoverageSet}(c) = \{ x, y, z \} \]

Case Competence: The Basics

- Case-Base Coverage
  - How does the coverage of the case-base depend on the coverage of its cases?
  - Unique & Redundant Coverage

A Competence Model

- Competence Groups
  - Maximal clusters of cases exhibiting shared coverage.
  - "Connectedness" of cases
A Competence Model

- Computing Competence Groups
  - \( c_1 \) & \( c_2 \) share coverage iff their case competence overlap
  - \( c_1 \), \( c_2 \) & \( c_3 \) are a Competence Group iff they share coverage

The Importance of Competence Groups

- Independent regions of coverage
  - \( \Rightarrow \) Independent competence contributions

Fundamental Unit of Competence?

Group Coverage - The Basic Idea

- Case Density \( \propto \) Coverage Redundancy \( \propto ^{-1} \) Group Coverage

- Regularity Assumption - Similar Problems \( \Rightarrow \) Similar Solutions
What is a Competence Hole?

- What is a competence hole?
  - Any uncovered region of the target space
- What makes a competence hole interesting?
  - Size of the hole
  - Relevance to target problems

Two Types of Competence Holes

Type 1
- Insufficient cases within the case-base.
- Lost coverage.

Type 2
- Due to domain constraints – impossible value combinations.
- No lost coverage

Identifying Interesting Holes

- Methodology
  - Competence groups that are close to each other may ultimately merge into a single group
  - The missing cases are competence rich spanning cases
  - Search for new spanning cases in the regions between nearby competence groups
Identifying Interesting Holes

- **Boundary Cases**
  - Each pair of groups $G$, $H$ has a corresponding pair of boundary cases, $g_H$, $h_G$ with maximal similarity.
  - Each group has a set of $n-1$ boundary cases corresponding to the $n-1$ other groups in the case-base.

Interesting Holes

- For each group we can search for new spanning cases between it and its nearest neighbour group.

Case Generation (Boundary Method)

- **Methodology**
  - Generate a new case from the feature values of the related sets of the boundary pair cases.
  - **Nominal Features**
    - Most frequent value
  - **Continuous Features**
    - Mean value
Discussion

- Applications
  - Authoring & Maintenance
- Related Work
  - Competence Categories (Smyth & Keane, IJCAI 1995)
- Assumptions & Applicability
  - Representational Biases

Case Authoring

- Current Tools
  - Poor modelling & visualisation techniques
- Authoring Guidance
  - Identification of redundant cases and inefficient groups of cases
  - Identification of regions of poor competence
  - Predictive measures of competence and performance

CASCADE

- Case Authoring Support & Development Environment
  - Traditional Authoring Functionality (defining & editing cases)
- Visualisation & Modelling Tools
  - Competence Groups (coverage & density analysis)
  - Competence Graphs
Group Size vs. Coverage

Related Work

- Competence Categories (Explanatory Model)
  - Pivotal, Spanning, Support, Auxiliary Cases
  - Coarse Grained Competence Patterns
- Current Model (Predictive Model)
  - Fine Grained Competence Measures
  - Above categories are found within competence groups. Eg, singleton competence groups hold pivotal cases.

Assumptions

- Representativeness
  - Case-base is a representative sample of the target problem space \(\Rightarrow\) tractable coverage estimates.
- Regularity & Uniformity
  - Density models assume that regions of the problem space are regular and uniform.
- Real World Case-Bases
  - If these assumptions do not hold then the quality of our competence predictions will degrade (gracefully?).
Conclusions

- A Competence Model for CBR
  - Positive initial results
- Future Work
  - Further experiments
  - Authoring & maintenance applications
  - Visualising case-bases