Using Domain Knowledge to Influence Similarity Judgments

Timothy Cain
Michael J. Pazzani
Glenn Silverstein

Department of Information and Computer Science
University of California, Irvine

Outline

- Research goals
- Case-based reasoning vs. Explanation-based learning
- Combination: CBR+EBL
- An example
- Analysis
- Related work
- Future research
- Summary

Research Goals

- Augment the CBR nearest neighbor algorithm so that similarity function makes use of domain knowledge
- Develop a general purpose knowledge-intensive CBR system that can be used on a variety of problems

Case Based Reasoning

Making predictions from precedents (similar prior cases).

1. Add prior cases to memory.
2. To make a prediction on new case
   A. Retrieve “most similar” case(s)
   B. Apply outcome of prior case to new situation.

Defining “similarity” in nearest neighbor:
Number of features in common - sensitive to representation

\+ Doesn’t require precise understanding of domain.

\+ Precedents can be meaningful to human expert

\- Not all precedents retrieved will be relevant to the new case

– Doesn’t take advantage of domain knowledge

Explanation-based Learning

Create new rules by combining preconditions of old rules.

1. Explain a case (using domain theory)
2. Remove those parts of the case not referenced in the explanation
3. The relevant features are those that were needed to create an explanation.

+ Not sensitive to representation (ignores irrelevant features)

- Results are only as accurate as existing knowledge.

Motivation: Combining EBL and CBR

Degradation:

+ CBR degrades gracefully: Predictive accuracy is a function of the distance between cases.
- EBL is brittle: Learned knowledge is set of necessary and sufficient conditions.

Attention:

+ EBL is knowledge-intensive. Domain knowledge can be used to determine the relevance of features.
- CBR is knowledge-free. Doesn't tolerate large numbers of irrelevant features.

Accuracy:

+ CBR is knowledge-free**.** Not misled by incorrect domain knowledge.
- EBL is knowledge-intensive. Only as accurate as human provided domain knowledge

Approach: Combining EBL and CBR

- Make the similarity function influenced by the domain knowledge.

$$\text{sim}(x,y)= \begin{cases} 1 & \text{if } x= y \\ 0 & \text{if } x \neq y \end{cases}$$

$$\text{CBR:} \quad \frac{\sum_{i=1}^{n} \text{sim}(\text{case}_i,\text{cue}_i)}{n}$$

A similarity function for Generalized Explanation-based Learning

$$\text{relevance}(\text{case}_i) = \begin{cases} 1 & \text{if case}_i \text{ is relevant} \\ 0 & \text{if case}_i \text{ is not relevant} \end{cases}$$

EBL:

$$\frac{\sum_{i=1}^{n} \text{relevance}(\text{case}_i) \times \text{sim}(\text{case}_i, \text{cue}_i)}{\sum_{i=1}^{n} \text{relevance}(\text{case}_i)}$$

Note: true EBL would return a 1 if all of the relevant features matched and return a 0 otherwise

A parametrized similarity function for EBL+CBR

β: Additional weight of a relevant match

$$\frac{\sum_{i=1}^{n} \mathrm{sim}(\mathrm{case}_i, \mathrm{cue}_i) + \beta \sum_{i=1}^{n} \mathrm{relevance}(\mathrm{case}_i) \times \mathrm{sim}(\mathrm{case}_i, \mathrm{cue}_i)}{n + \beta \sum_{i=1}^{n} \mathrm{relevance}(\mathrm{case}_i)}$$

$\beta = 0$, normal CBR

$\beta > 0$, relevant features weighed more heavily

The CBR + EBL system: Storing Cases in Memory

Given: A set of cases with known outcomes
A domain theory

Do: For each case, explain its outcome.
Gather relevant features.

Output: A set of cases, each with a set of relevant features

Note: 1. Cases with the same outcome can have different sets of relevant features
2. Cases that can't be explained have no relevant features

The CBR + EBL system: Retrieval

Given: A "cue" case with an unknown outcome
Value for β

Do: For each case in memory, compute similarity to cue.

Output: A list of similar cases, sorted by similarity

The CBR + EBL system: Explanation

Given: A case and an outcome

Do: Use domain theory to explain outcome

Output: Explanation for the case

Some of the 83 Case Features

SUBSIDNO is industry subsidized by opponent
BEFORELC number of months before elections at start
PRELECT number of months after elections at start
POSTELCT number of months before elections at end
ELECTEND number of months after elections at end
RULING ruling party's perception of strength at end
LEVL_OPP level of political influence of opposing interest groups
LEVL_SUP level of political influence of supporting interest groups
PRESSCOV press coverage in opponent nation
MIN1_POS position of ministry most often quoted
MIN2_POS position of ministry second most often quoted
MIN3_POS position of ministry third most often quoted
VISIT time from end of negotiations until govt head and US Pres. meet
HDVISIT time before end of negotiations until govt head and US Pres. meet
US_CONG is opponent nation focusing on Congress as pressure or obstacle
NEWS position of major economic newspaper in opponent nation
MAJNEWS position of major newspapers in opponent nation
UNI_ACT would unilateral action in GATT invoke compensation

Classification of cases by final agreement

Given: A set of cases

Do: For each case, calculate an outcome based on final trade agreement (FTA):

uswin:	FTA near US closing bid
oppwin:	FTA near Opponent closing bid
noagg:	no FTA
tie:	otherwise

Output: A set of cases, each with an outcome

Domain Theory

```
  % the opponent will win if the industry involved is a traditional one,
   % there are 2 or less opponent ministries involved, the opponent has
   % rarely conceded on this particular commodity, and the commodity was
   % discussed in GATT. Optionally, strong anti-us interest groups can
   % help the opponent to win, too.

oppwin(Case) :-
        opt(strong_opponent_groups(Case)),
        traditional_industry(Case),
        few_involved_opponent_ministries(Case,2.5),
        few_opponent_concessions_on_commodity(Case,1),
        commodity_negotiated_in_GATT(Case).

traditional_industry(Case) :-
   var_comp(Case,indexist,<,1.5).

very_pro_us_factors(Case) :-
        opp_industry_not_at_maximum(Case),
        industry_not_subsidized(Case).

industry_not_subsidized(Case) :-
   subsidno (Case,0).
```

Classification of cases by domain theory

Classification	US factors	Opponent factors
uswin:	strong	weak
oppwin:	weak	strong
tie:	weak	weak
noagg:	strong	strong

Accuracy of Domain Theory

Testing Procedure: If the domain theory predicts exactly one outcome, and that outcome is identical to the observed outcome, then the domain theory makes a correct prediction.

Percent of 50 cases predicted correctly: 44%

Sources of error:

1. Domain theory produced by computer scientists.
2. Domain not as precise as many studied by expert systems
3. Imprecise classification of cases (uncertainty of positions)
4. Uncertainty of some data
5. Errors in coding data

An Example

case16: Trade negotiations involving air routes between February and May of 1985. The negotiations were terminated without an agreement.

case21: semiconductor and office machinery negotiations that started in May of 1985 and ended in July of 1986 with no agreement.

case31: trade negotiations involving telecommunications and radio equipment between June, 1985 and January, 1986. The US achieved most of its stated goals in this case, while the trading partner's position changed substantially.

Retrieval (finding cases similar to case16)

$\beta = 0$

```
case31     uswin
case10     noagg
case21     noagg
case4      uswin
```

$\beta = 10$ (similar results obtained for $\beta > 1$)

```
case21     noagg
case10     noagg
case11     tie
case7      tie
```

Evaluation of CBR+EBL

"Leave-one-out" testing procedure: Use 49 cases to predict outcome of 50th case. Repeat 50 times each time testing on a different case.

Test case	**Percent Classified Correctly**
CBR Alone (α =1, β = 0)	52%
CBR+EBL (α =1, β = 15)	70%
Domain Theory Alone	44%

Analysis of CBR+EBL

Domain theory improves CBR accuracy by 18%

- Irrelevant features not correlated with outcome hurt CBR

CBR improves domain theory accuracy by 26%

- Correctly explained cases favorably influence CBR
- A large number of similarities among falsely classified irrelevant features compensates for incorrect domain theory
- CBR compensates for domain theory that predicts multiple outcomes. CBR reasons from what has occurred; domain knowledge represents what is expected to happen.

Related Work

- Explanation-based indexing (Barletta and Mark)
  - uses primary and secondary indices on features
- Optimist (Clark)
  - has rules to determine relevant features and their weights
- IB4 (Aha)
  - learns weights for each feature

Future Research

- Using inductive learning methods to improve accuracy of domain theory
- Continuous similarity function
- Individual feature weights (a set of β values)
- Weight learning

Summary

1. Constructed general CBR+EBL system that accepts case descriptions and domain theory as data

2. Tested system on 50 cases of actual trade negotiations

3. Demonstrated that CBR+EBL can result in more accurate predictions than either method applied individually.

Sensitivity to β values