DATA ENRICHMENT IN CASE-BASED REASONING FOR SOFTWARE COST PREDICTION

Rahul Premraj, Martin Shepperd and Michelle Cartwright {rpremraj, mshepper, mcartwri} @bournemouth.ac.uk
ESERG, School of Design, Engineering and Computing, Bournemouth University, United Kingdom

Key words to describe this work: software engineering, case-based prediction, cost estimation, meta-data

Key Results: Cases are often indiscriminatingly added to case bases. This potentially results in poor solutions proposed by 'misleading' cases. Research to-date has resulted in development of a prototype that can potentially filter 'misleading' cases in case bases to avoid delivery of poor solutions. The prototype was tested in a pilot study, in which absolute residuals (error in prediction) were reduced by 45.2% from 68.17 hours to 37.33 hours.

How does the work advance the state-of-the-art? Our research explores methods to enrich data sets that use feature vectors for data representation. We identify the limitations of such representation and propose two methods to address deficiencies. Though the research is specific to software engineering data sets, the proposed techniques can be easily adapted to suit other problem domains that use similar forms of data representation.

Motivation (Problems addressed): Feature vectors pose constraints on the depth and richness of information about the cases in case bases. The lack of comprehensive knowledge leads to application of simplistic adaptation routines for case modification in case-based reasoning systems. Hence, proposed solutions are sub-optimal. This is a generic problem that can be observed in data sets across a variety of domains.

Introduction

The importance of estimating costs of a software project is crucial since solutions to many business and technical questions that need to be addressed at an early stage, are based upon this information. Examples of such questions are the decision whether to undertake a project or not, need to hire more staff or whether profits can be increased by outsourcing?

The requirement of having an accurate estimate of costs has triggered research in the field for more than 30 years. Proposed models and techniques have been largely algorithmic in nature (e.g. COCOMO) however, such models have not been generally effective. More recently, a few software engineering research groups have applied case-based reasoning (CBR) to generate software estimates.

The CBR [1] methodology is analogous to the manner of problem solving by experts. On being presented a problem, CBR systems search a case base, which consists of episodes of past problem solving experiences, to search for a case that is most *similar* to the problem (*retrieval*). The solution or solution deriving process is extracted from the retrieved case and used to propose a solution for the problem. The proposed solution may be adapted to adjust the solution for dissimilar features in the problem and retrieved case (*adaptation*).

Results have suggested CBR to be a promising technique for cost estimation but further research is needed to optimise its application [2]. A major constraint is the nature of software engineering (SE) data sets that largely limit the adaptation routines to be relatively naïve, which results in sub-optimum predictions. Given a richer data set, CBR systems will have more knowledge at their disposal to utilise to adjust for non-matching features and propose an adapted solution. Hence, a natural direction to research is to refine and enrich SE data representation to provide for the applications of more complex decision making routines. This opens doors to apply more sophisticated routines on available data to improve performance.

Software Engineering Data sets

Predominantly, SE case bases, comprise of software projects represented using feature vectors [3]. In such data sets, some features represent the problem state and the remaining features describe the solution state. In SE data sets, all except one feature describe the problem and are used to compute similarity between the problem and cases in the case base. The remaining feature (cost in our case) is the proposed solution. It is the case base that forms the backbone of any CBR tool since their structure and contents determine the quality of solutions proposed by the system.

Case representation using feature vectors eases the task of building, storing and maintaining case bases. They also enable the employment of simple similarity measuring and retrieval techniques. However, such representation constrains the depth and nature of data

represented that may potentially lead of loss of vital expert knowledge. In such data sets, it has not been possible to highlight relationships between recorded features that may potentially be reflected upon during problem solving. Also, there is no provision for storing data that has not been covered or represented by the feature vectors. As a result, adaptation (adjusting proposed solution to suit the problem) routines are largely limited to simple adjustments such as simple average, distance weighted average or linear adaptation [4].

Data Set Enrichment

To address the deficiencies of SE data sets, we propose their refinement in two ways. Firstly, we propose enriching data sets with meta-data that reflect the quality of cases individually in the case base. From our observation of SE data sets, we take note that almost no data set gives any information about the quality of individual cases. Such information will be helpful to decide if the retrieved case is worth using or should be discarded.

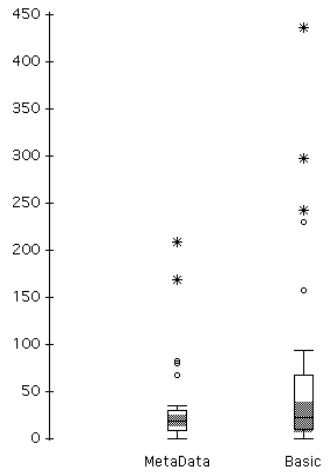
A pilot study [5] was conducted to establish the worth of introducing data into the case base which helps selectively reject retrieved cases that have shown to consistently propose poor quality solutions in the past. Our proposed model evaluated the error in prediction and associated two contrasting ranks (based on the error in prediction) with the retrieved case. These ranks were updated on every instance of retrieval for the retrieved case. Cases were unconditionally retrieved and used on their first retrieval. However, before being reused in following estimations, the ranks of the retrieved case were analysed by a fuzzy model, which judged whether the case should be reused or discarded. If discarded, the next nearest case approved by the fuzzy model was used for estimation. The pilot study used a small data set comprising of web projects that was injected with misleading seed cases. The proposed model successfully identified 8 seed cases that were later rejected to be reused in 8 different estimations. The remaining seed cases were never retrieved and hence, not identified. The Figure 1 plots absolute residuals from the pilot study and a basic run (without meta-data). The results warrant further research into our proposed model to identify poor cases in the case base to increase the estimation accuracy and quality of the case base.

Secondly, we plan to refine the logical or conceptual representation of data in the case base by incorporating object-oriented (O-O) concepts. O-O concepts will help represent cases like in the real world and highlight relationships between features of the cases. Such hierarchical representation will help breakdown cases that can result in effective retrieval and adaptation. Explicit representation and mapping of underlying relationships between features will ease the process of knowledge elicitation that can be exploited during adaptation. Also, dealing with missing or inadequate data with the help of several layers of abstraction is another related area that can be exploited.

Conclusions

Our research to-date shows that enriching the structure and contents of the data sets can result in an increase in estimation accuracy in our problem domain. We plan related future work to increasing performance of CBR systems. Importantly, though our research focuses upon the application of CBR to software engineering, it can be easily adapted to other problem domains that use data sets of similar nature.

Figure 1: Box plots comparing absolute residuals of runs with and without using meta-data



References

1. Aamodt, A. and E. Plaza, *Case-Based Reasoning: Foundation Issues, Methodological Variations, and System Approaches.* AI Communication, 1994. **7**(1): p. 39-59.
2. Shepperd, M. and C. Schofield, *Estimating software project effort using analogies.* IEEE Transactions on Software Engineering, 1997. **23**: p. 736-743.
3. Spalazzi, L., *A survey on Case-Based Planning.* AI Review, 2001. **16**(1): p. 3-36.
4. Kirsopp, C., E. Mendes, R. Premraj, and M. Shepperd. *An Empirical Analysis of Linear Adaptation Techniques for Case-Based Prediction*. in *ICCBR 2003*. Trondheim, Norway: Springer.
5. Premraj, R., M. Shepperd, and M. Cartwright. *Meta-data to Guide Retrieval in CBR for Software Cost Prediction.* in *8th UK Workshop on CBR*. 2003. Cambridge, UK.

DATA ENRICHMENT IN CASE-BASED REASONING FOR SOFTWARE COST PREDICTION

RAHUL PREMRAJ, MARTIN SHEPPERD & MICHELLE CARTWRIGHT

{ rpremraj, mshepper, mcartwri } @ bournemouth.ac.uk

EMPIRICAL SOFTWARE ENGINEERING RESEARCH GROUP

http://dec.bournemouth.ac.uk/ESERG/ESERG.html

PROBLEM DOMAIN

Cost estimation in the early stages of a software project is recognised as a vital task in the field field of software engineering. Costs can be a crucial determinant of project plans and might potentially drive or influence the quality of the end product. Many business and technical questions that need to be addressed at an early stage, are based upon this information. Examples of such questions are the decision wheter to undertake a project or not, need to hire more staff or wheter profits can be increased by outsoucing?

A TYPICAL SOFTWARE ENGINEERING DATASET

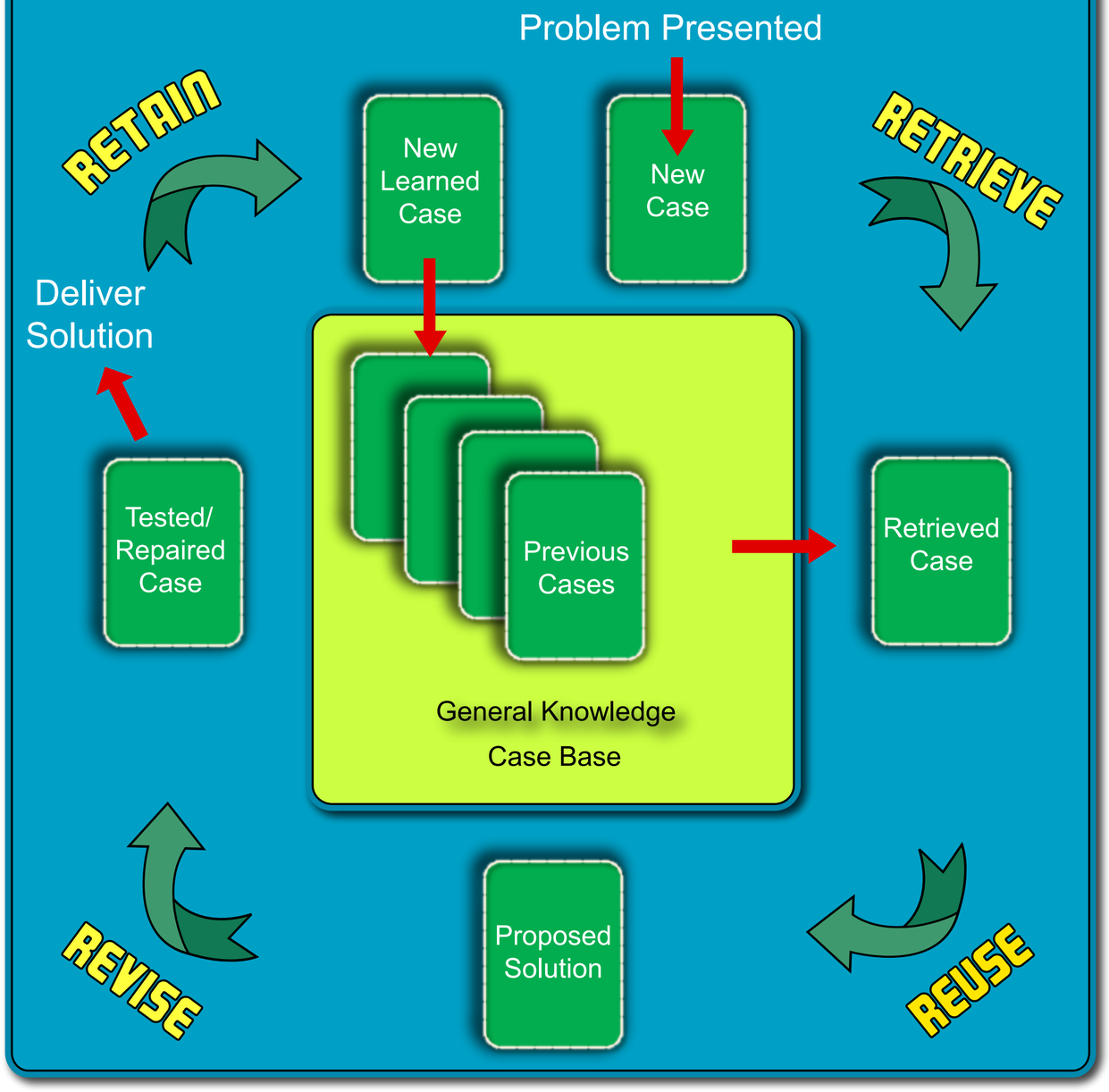
FEATURE VECTORS

	A	S	T	U	V	W	X	Y
	YK	ARDWARE.NAME	NGUAGES.NAM	DATESTART	DATEEND	Duration	SIZE_ep	WORK
35	920901	(mf+pc, mini+pc,	def	18-Jan-99	30-Jun-99	5	216	
36	920901	(mf+pc, mini+pc,	def	26-Mar-99	28-Oct-99	10	293	
37	920901	Mainframe	PL/I	01-Sep-98	17-Mar-99	6	442	
38	920901	Mainframe	PL/I	01-Jun-99	20-Aug-99	2	279	
39	920901	Mainframe	PL/I	16-Aug-99	15-Oct-99	2	284	
90	920901	Mainframe	PL/I	16-Aug-99	01-Jan-00	5	404	
91	920901	Mainframe	PL/I	16-Aug-99	01-Jan-00	5	404	
92	920903	PC	Visual Basic 5	16-Aug-92		5	138	
93	920903	Mainframe	COBOL	01-May-90		22	550	
94	920903	Networked	SQL-Windows	01-Nov-91		4	291	
95	940104	(mf+pc, mini+pc,		01-Jan-94		4	406	
96	940104	(mf+pc, mini+pc,		01-Nov-92		25	2876	
97	940104	(mf+pc, mini+pc,		01-Apr-92		31	2531	
98	940104	(mf+pc, mini+pc,		01-Aug-93		3	521	

Pros: Easy to build, store and maintain case bases.

Cons: Constraints on the depth and nature of information, potential lose of expert knowledge, not been possible to highlight relationships between recorded features.

CASE-BASED REASONING CYCLE



RESEARCH FOCUS

A major constraint is the nature of software engineering data sets that limit the adaptation routines to be relatively naïve, which results in sub-optimum predictions. Given a richer data set, CBR systems will have more knowledge at their disposal to utilise to adjust for non-matching features and propose an adapted solution. A natural direction to research is to refine and enrich SE data representation to provide for the applications of more complex decision making routines. This opens doors to apply more sophisticated routines on available data to improve performance.

EMBEDDING META-DATA

We propose enriching data sets with meta-data that reflect the quality of cases individually in the case base. Such information will be helpful to decide if the retrieved case is worth using or should be discarded.

T	U	V	W	X
GES.NAM	DATESTART	DATEEND	Duration	SIZE_ep
	18-Jan-99	30-Jun-99	5	216
	26-Mar-99	28-Oct-99	10	293
	01-Sep-98	17-Mar-99	6	442
	01-Jun-99	20-Aug-99	2	279
	16-Aug-99	15-Oct-99	2	284
	16-Aug-99	01-Jan-00	5	404
	16-Aug-99	01-Jan-00	5	404
Basic 5	16-Aug-92		5	138
L	01-May-90		22	550
Vindows	01-Nov-91		4	291
	01-Jan-94		4	406
	01-Nov-92		25	2876
	01-Apr-92		31	2531

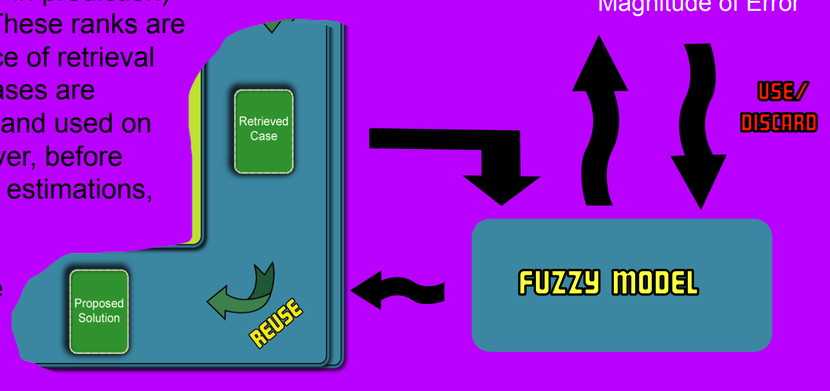
	A	B	C
1	Rank	Frequency	BMMRE
2	0.1471	2	0.9657
3	0.4155	11	0.3568
4	0.3250	3	0.1402
5	0.5361	6	0.3724
6	1.0000	1	0.6914
7	1.0000	1	0.6563
8	0.3021	4	0.1414
9	0.7778	1	0.2819
10	0.1755	4	0.0960
11	0.7500	1	0.4500
12	0.4241	7	0.3318
13	0.4724	6	0.3787
14	0.3975	3	0.2869
15	1.0000	1	0.8725
16	0.3987	6	0.4084
17	0.9231	1	0.9346
18	0.7143	1	0.4528

Frequency

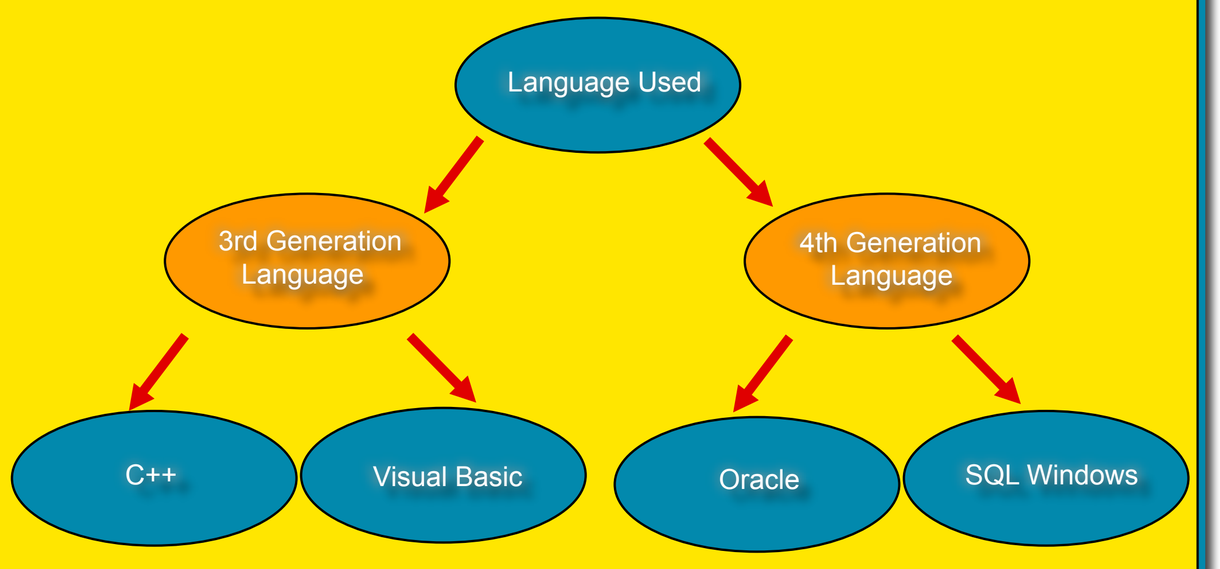
Rank

Balanced Mean Magnitude of Error

Our proposed model evaluates the error in prediction and associated two contrasting ranks (based on the error in prediction) with the retrieved case. These ranks are updated on every instance of retrieval for the retrieved case. Cases are unconditionally retrieved and used on their first retrieval. However, before being reused in following estimations, the ranks of the retrieved case are analysed by a fuzzy model, which judge whether the case should be reused or discarded.



OBJECT-ORIENTED REPRESENATION



- Helps represent data in a more organised fashion
- Relationships between features can be created and used during problem solving
- Explicit representation and mapping of underlying relationships between features will ease the process of knowledge elicitation that can be exploited during adaptation
- Dealing with missing or inadequate data with the help of several layers of abstraction can be exploited.