Investigating effort prediction of web-based applications using CBR on the ISBSG dataset

Sukumar Letchmunan Marc Roper Murray Wood
Dept. Computer and Information Sciences
University of Strathclyde
Glasgow, U.K.
Sukumar.Letchmunan@cis.strath.ac.uk *Marc.Roper@cis.strath.ac.uk* *Murray.Wood@cis.strath.ac.uk*

As web-based applications become more popular and more sophisticated, so does the requirement for early accurate estimates of the effort required to build such systems. Case-based reasoning (CBR) has been shown to be a reasonably effective estimation strategy, although it has not been widely explored in the context of web applications. This paper reports on a study carried out on a subset of the ISBSG dataset to examine the optimal number of analogies that should be used in making a prediction. The results show that it is not possible to select such a value with confidence, and that, in common with other findings in different domains, the effectiveness of CBR is hampered by other factors including the characteristics of the underlying dataset (such as the spread of data and presence of outliers) and the calculation employed to evaluate the distance function (in particular, the treatment of numeric and categorical data).

Keywords: Effort estimation, case-based reasoning, web applications.

1. INTRODUCTION

With the increasing prevalence of web-based applications comes the requirement for the accurate estimation of the development costs associated with such applications. Accurate estimates are essential for companies to make competitive bids in the market and to efficiently resource development projects. In spite of considerable research in this area, no prediction techniques have proven to be consistently accurate. However, case-based reasoning (CBR) has been shown to be one of the stronger performing techniques, although this is usually in the context of traditional rather than web-based applications.

This paper investigates the application of CBR to a set of web-based application data. Web applications are typically characterised by shorter development cycles, smaller teams, a variety of programming languages or frameworks, and less formal process control and estimation strategies (Reifer [11] contains a far more comprehensive comparison). The main purpose behind the study is to investigate the optimal number of analogies (i.e. how many of the most similar cases should be taken into account) to employ when making an estimate, but the aim is also to note any other issues that arise when using CBR on web application data.

The main findings of the paper are that there is no consistent improvement in the accuracy of the estimates as the number of analogies increases. This is shown to be due to a number of factors including the similarity measure employed which appears to bias numeric data, and the distribution of data – in particular the presence of small or large values and a lack of an even distribution of values.

2. RELATED WORK

Recently, research on cost estimation for web applications has started attracting more attention. Various modeling techniques drawn from statistics, machine learning and knowledge acquisition have been used with various degrees of success and on a limited number of mostly small and medium size data sets [2, 4, 16].

Cost estimation methods can be divided into two methods: non-model based (expert knowledge) and model based estimation methods (cost estimation tools). Combinations of both approaches are known as composite methods [12].

Non-model based estimation methods require the heavy involvement of experts to generate an estimate of the new project [19]. The estimate will be based on the experts' accumulated experience rather than any particular model.

Model-based, or algorithmic, estimation methods are not dependent on individual's capabilities but require past project data for model building. Examples are OLS

regression, the Constructive Cost Model (COCOMO) [20], and Classification and Regression Trees (CART). The major drawback with this model-based approach such as example in COCOMO provides equations that incorporate system size as the principal effort driver. Predicted development effort is then adjusted to accommodate the influence of 15 additional cost drivers. The main conclusions were these models perform poorly when applied to other environments [18].

In the recent years machine learning techniques have been used as a complement or alternative to the previous categories. There are a variety of machine learning methods including: artificial neural networks [4], rule induction algorithms [2], case based reasoning [9, 13, 16, 17], hybrid approaches such as neuro-fuzzy methods and multiple learners [14].

The machine learning approach has been explored most in the context of cost estimation is that of CBR. Case based reasoning (CBR) was first formalised in the 1980s following from the work of Schank and others on memory, and is based upon the fundamental premise that similar problems are best solved with similar solutions. CBR is based on the psychological concepts of analogical reasoning, dynamic memory and the role of previous situations in learning and problem solving. Basically a CBR processing cycle is composed of four stages [1]: (1) Retrieve the most similar project case; (2) Reuse the project to attempt to solve the problem; (3) Revise the suggested solution if necessary; (4) Retain the solution and the new problem as a new project.

The appeal of CBR may rest on the fact that users may be more willing to accept a solution from a form of reasoning which is more akin to human problem solving, and even though there is no single best software cost estimation model, CBR is rated among the best methods in a variety of circumstances [12]. In addition to being intuitive and having a reasonable level of accuracy, CBR is also simple and flexible, and may be applied to both qualitative and quantitative data, reflecting typical industrial datasets [8].

CBR also has some disadvantages. As with algorithmic models, the effect of old data points is not clear. As an organization develops and successively introduces new technology, the older data points may become increasingly irrelevant and potentially misleading [13, 16]. This needs more investigation, especially in the area of web applications, as there has been a rapid change in term of languages and technologies even in the short time that such systems have been around.

There are also a number of challenges towards the effective application of CBR, some of which are general to a domain and others which may only be relevant to a particular dataset. The problems that most researchers encounter in applying CBR fall into the following categories [15]:

(i) Feature Subset Selection

There are many features in the dataset but not all of them are necessarily relevant for predicting the project effort. They might be redundant or error data.

(ii) Scaling

Scaling or standardization represents the transformation of attribute values according to a defined rule such that all attributes are measured using the same unit. Angel for example assigns zero to the minimum observed value and one to maximum observed value.

(iii) Similarity Measure

A distance measure in CBR is the degree of similarity between two projects in terms of their effort drivers. Euclidean distance is most commonly used to solve this problem. Similarity measures for categorical data typically employ a value of 1 to represent a match and 0 otherwise. This is an interesting point that demands further investigation.

(iv) How Many Analogies To Use

The number of analogies refers to the number of most similar cases that will be used to generate the estimation. Most of the previous work employs 1, 2 and 3 analogies, but there is no clear rule on how many analogies to be use [6, 9, 10].

(v) Analogy Adaptation

Analogy adaptation concerns how to generate the estimation once the analogies are retrieved. Different approaches include using the mean of analogies or nearest neighbour.

Several papers have investigated this last aspect in detail [6, 16], focusing on dataset size as one of the major factors concerning the accuracy of analogy based methods by analyzing the trends in estimation accuracy as the datasets grow. Although the work of Kadoda et al. confirmed that analogy based estimation achieves better results by employing larger training sets [6], Shepperd and Schofield claim that accuracy in analogy based estimation does not always increase within the number of projects or datasets – showing instead that it can be affected greatly by introduction of outlying projects [16].

As discussed above, the question “Does accuracy improve as the number of projects cases increased?” is still in doubt. Much of the work in this uses public datasets, many of which are old and not employing

web application data. Therefore it may be fruitful to investigate this question by using a web application dataset.

3. THE DATASET

The investigations in this paper are all based upon the International Software Benchmarking Group (ISBSG) Release 10 dataset [5]. The data in ISBSG repository have come from over twenty-five countries, with 60% of projects being less than 7 years old. Software practitioners voluntarily submitted the projects in the ISBSG data set which was collected using questionnaire. The ISBSG collection pays much attention to the quality of gathered data. There are special data validation forms and the project managers are asked to report the confidence the have in the information they have provided [21]. A specific field is used containing a rating code of A, B, or C applied to the project data by the ISBSG quality reviewers to denote the following:

A= The submission satisfies all the criteria for seemingly sound data.

B= The submission appears fundamentally sound but there is some evidence to question some of the supplied data.

C= The submission has some fundamental shortcomings in the data.

As ISBSG point out, in any statistical analysis only projects with A and B rating should be used. Of the 4,106 project summaries in the repository, 422 are related to web applications, and it is this subset which is the subject of this study.

The dataset covers a wide range of applications, development techniques and tools, languages and platforms. Of the total of 109 features that may potentially appear in the ISBSG dataset, just 9 were selected which are considered relevant to this work, or which could potentially have an impact on effort. The table below lists the features used in this study.

***Table 1**. Description of selected features*

Name	Description
Case Name	*Index*
CountApproach	*Counting approach that been used such as IFPUG, LOC*
WorkEffort	*Summary of work effort in hours*
DevType	*Development Type*
AppType	*Application Type*
PriProgLang	*Primary Programming Language*
Database	*Ist Database system*
FunctionalSize	*Functional Size*
AdjustedFP	*Adjusted Function Points Count*

4. METHODOLOGY

The main aim of this paper is to investigate the impact of the number of analogies on the accuracy of estimates obtained through case-based reasoning. Consequently, the large dataset needs to be broken down into smaller subsets in order to provide more opportunities to experiment with using different numbers of analogies, and also to mimic more closely the data set size that is likely to be available in an industrial context. The 422 web application records in the ISBSG dataset were divided into 3 groups, each consisting of 67 unique records (cases). Care was also taken not to include any cases that are incomplete. Similarly to previous studies (e.g. [6]), in order to explore the impact of the number of cases, these three datasets are further subdivided (randomly again) to populate smaller datasets consisting of 17, 33, and 47 records. This exercise yields a total of twelve data sets: three initial groups (labelled G1, G2 and G3) each containing 67 cases, each randomly subdivided into groups of 13, 33, 47 and labelled G1-Ran1-17, G1-Ran1-33, G1-Ran1-47, G1-Ran1-67, G2-Ran1-17, G2-Ran1-33, … G3-Ran1-67. This procedure is then repeated a further two times to guard against any freak results introduced by the randomising process [2] producing a second (G1-Ran2-17, G1-Ran2-33, … G3-Ran2-67) and third (G1-Ran3-17, G1-Ran3-33, … G3-Ran3-67) – thirty-six data sets in all[1].

The CBR tool Angel [16] was used for this experiment to determine the prediction value of the effort according to jack-knife method (also known as leave one out cross- validation). This procedure is the same as that adopted by others, including Mendes et al. [10], and follows the procedure outlined in below. This was applied to all 36 datasets.

For each case in the data set:
- *Discard the effort data for that case (marked as "unconfirmed" - in order to simulate a new project)*
- *Using from 1 to 7 analogies:*
  - *Use the remaining cases to estimate the effort for the unconfirmed case*
- *Restore the original effort value for the unconfirmed case and return it to the dataset*

In Angel tool similarity is defined as Euclidean distance in n-dimensional space where n is the number of project features. Each dimension is standardized so all dimensions have equal weight.

5. RESULTS AND ANALYSIS

To gauge the accuracy of each estimated effort value two values are calculated for each number of analogies(k) used for each dataset: the Mean Magnitude of Relative Error[2] (MMRE), and PRED(25)[3] [22]. It is considered that good prediction models should exhibit MMRE values of up to 25%, and

[1] Note that GNRanM-67 will be identical for all values of M, but are included in the results for the purposes of comparison.

2 The average MRE for each dataset, where the MRE is defined as |actual – estimate|/actual.

3 The percentage of projects that have an MRE value of <= 0.25.

PRED(25) values of at least 75%. The results for the MMRE are shown below. It has not been possible to include those for PRED(25) for reasons of space. The results are shown graphically in figures 1 to 9, where the number of analogies (k) is shown on the x-axis and the value of MMRE on the y-axis.

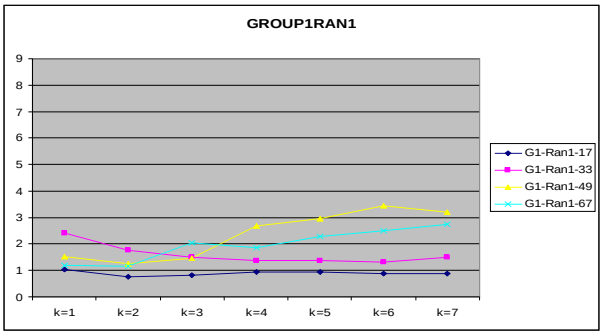

Fig.1. *Result of MMRE vs Analogies on Group1Ran1*

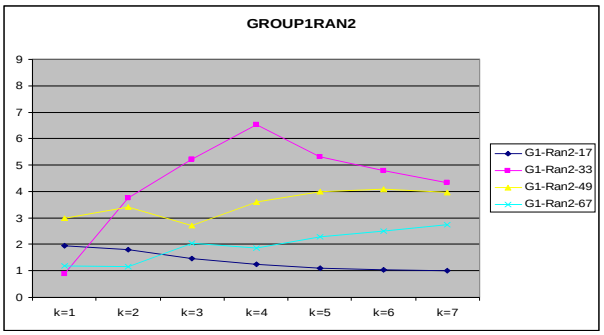

Fig.2. *Result of MMRE vs Analogies on Group1Ran2*

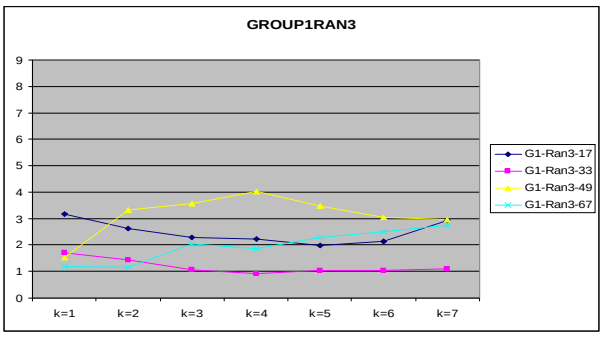

Fig.3. *Result of MMRE vs Analogies on Group1Ran3*

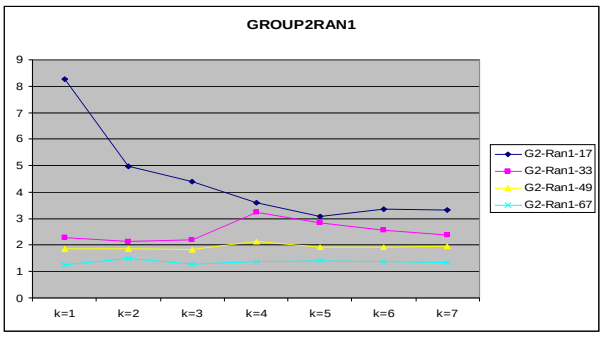

Fig.4. *Result of MMRE vs Analogies on Group2Ran1*

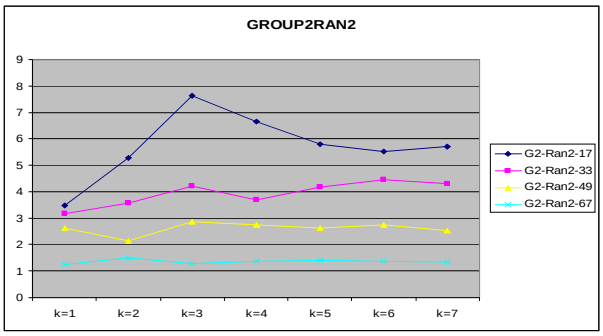

Fig.5. *Result of MMRE vs Analogies on Group2Ran2*

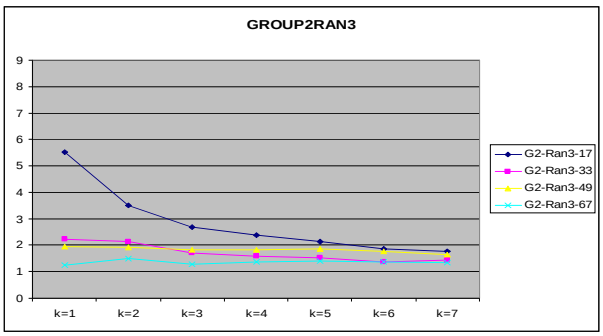

Fig.6. *Result of MMRE vs Analogies on Group2Ran3*

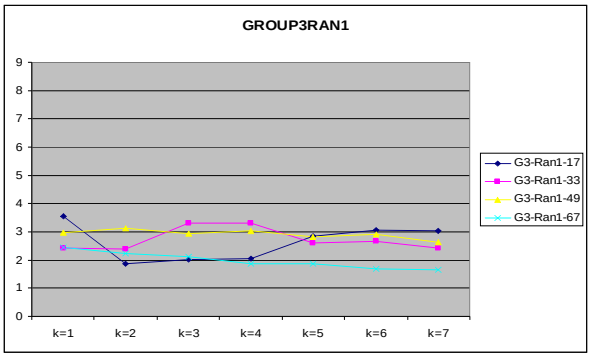

Fig.7. *Result of MMRE vs Analogies on Group3Ran1*

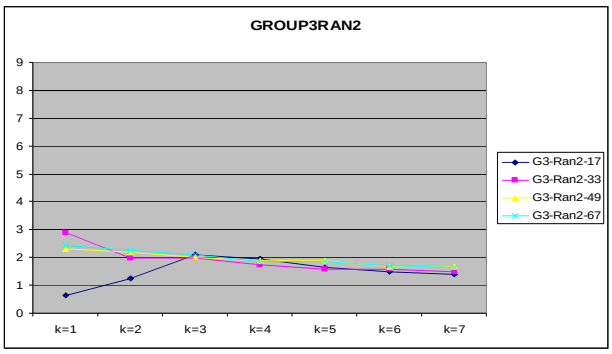

Fig.8. *Result of MMRE vs Analogies on Group3Ran2*

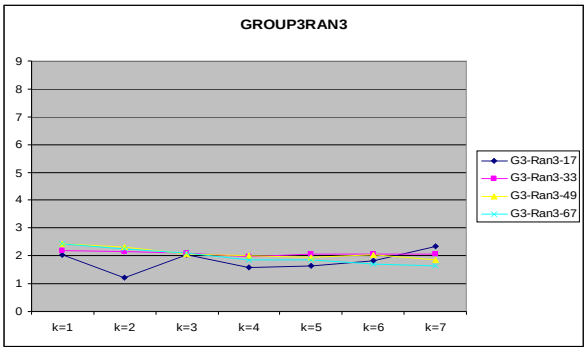

Fig.9. *Result of MMRE vs Analogies on Group3Ran3*

There are two immediately notable results concerning the MMRE values. Firstly, none of the averages is anywhere near the 25% value – in fact values below 100% are rare. Secondly, the graphs typically do not display any common trends. In some cases there is a general lowering of the MMRE values as k (the number of analogies) increases (for example Group2Ran3, which shows a gradual convergence as k gets larger), whilst other cases show completely the opposite trend, and others still display sudden peaks or troughs. The remainder of this section will attempt to provide an explanation for some of these more pronounced patterns by considering some particular questions.

5.1. What is the reason for the peak in the results for G1Ran2-33?

As can be seen, the results for this set show a very different pattern compared to G1Ran1-33 and G1Ran3-33 (drawn from the same set of 67 cases) and even for other configurations of the Group1 data (a similar shape can be observed in G1Ran3-49, but the peak value is considerably lower). Also, it is unusual that the MMRE value starts of as one of the lowest for k=1 and climbs to one of the highest for k=4. To investigate this result in more detail it is necessary to look more closely at the dataset (up to k=4 for space reasons), shown in table 2.

As can be seen for k=1, the most frequently predicted effort value is 47. This can be examined in more detail by looking at two different cases (those named13700 and 10566) which have very different values of actual effort (352 and 8580 respectively) but which show the same predicted effort value of 47 when k=1.

G1-Ran2-33		k=1	k=2	k=3	k=4
Case Name	**Actual Effort**	**Pred. Effort**	**Pred. Effort**	**Pred. Effort**	**Pred. Effort**
15720	*934*	*2240*	*1688*	*1225*	*988*
15008	*626*	*47*	*351*	*486*	*444*
13034	*4295*	*352*	*1621*	*3941*	*4361*
14779	*2891*	*2240*	*3267*	*5038*	*5184*
11100	*2240*	*2891*	*3593*	*5255*	*5346*
11648	*1056*	*1136*	*936*	*1935*	*1539*
10180	*2340*	*352*	*2986*	*4851*	*4712*
15440	*301*	*1136*	*707*	*782*	*691*
13127	*7496*	*352*	*199*	*1325*	*3836*
11283	*410*	*543*	*480*	*462*	*426*
15444	*2504*	*9231*	*10301*	*9366*	*7918*
14260	*3576*	*11372*	*9976*	*6666*	*6873*
15137	*543*	*410*	*364*	*382*	*393*
10358	*737*	*352*	*2143*	*1780*	*1619*
10427	*11372*	*8580*	*4313*	*5374*	*4924*
12078	*54*	*36*	*30*	*331*	*532*
11421	*36*	*24*	*39*	*337*	*537*
11132	*278*	*301*	*718*	*790*	*697*
13369	*418*	*425*	*417*	*371*	*414*
13700	*352*	*47*	*3771*	*2766*	*2231*
15603	*756*	*626*	*489*	*341*	*420*
14487	*3116*	*2240*	*1259*	*978*	*840*
13319	*47*	*352*	*4466*	*5476*	*6950*
12408	*425*	*418*	*414*	*368*	*412*
11718	*3934*	*47*	*4313*	*6666*	*5184*
13744	*1136*	*1056*	*704*	*715*	*1519*
13896	*1136*	*47*	*4313*	*6666*	*5075*
15468	*9231*	*11372*	*9976*	*7485*	*7488*
14911	*319*	*47*	*351*	*415*	*467*
11730	*5621*	*8580*	*5460*	*5071*	*4526*
13254	*24*	*36*	*45*	*341*	*540*
10566	*8580*	*47*	*5709*	*5680*	*4845*
11809	*655*	*47*	*336*	*330*	*437*

Table 2*. Predicted effort for G1-Ran2-33*

Each entry in the dataset conforms to the following format:

> Case Name, Count Approach, Summary Work Effort, Development Type, Application Type, Primary Programming Language, First Database System, Functional Size, Adjusted Functional Points.

For case name 13700 which holds the following data:

> 13700, IFPUG, 352, Enhancement, Process Control, ASP, SQL SERVER, 133, 133

the nearest calculated data points are:

- Rank 1, Distance: 0.654

  13319, IFPUG, 47, New Development, other: Sales contact management, ASP, ORACLE, 113, 113

- Rank 2, Distance: 0.713

  13127, IFPUG, 7496, New Development, Workflow support & management, ASP, SQL Server7, 786, 786

- Rank 3, Distance: 0.755

  15603, IFPUG, 756, Enhancement, Financial application area, Java, Interactive, 124, 124

- Rank 4, Distance: 0.755

  15008, IFPUG, 626, New Development, Financial application area, Java, Interactive, 116, 116

For case name 10566 which holds the following data:

- 10566, IFPUG, 8580, New Development, Financial transaction process/accounting, SQL, Oracle, 359, 359

The nearest calculated data points are:

- Rank 1, Distance: 0.663

  13319, IFPUG, 47, New Development, other: Sales contact management, ASP, ORACLE, 113, 113

- Rank2, Distance: 0.689

  10427, IFPUG, 11372, New Development, Financial transaction process/accounting, SQL, ORACLE, 859, 859

- Rank 3, Distance: 0.755

  11730, IFPUG, 5621, Enhancement, Document management; Financial transaction process/accounting Image video or sound processing, COBOL, IDMS-DB, 344, 344

- Rank 4, Distance: 0.756

  10180, IFPUG, 2340, New Development, Financial transaction process/accounting, Visual Basic, SQL-Server, 309, 309

In this example it appears that the similarity measure used in the Angel tool is having an effect on the prediction. Consideration of case 10566 suggests that the best fit (and highest rank) should be case 10427 as it has several of the categorical fields in common (Development Type, Application Type, Primary Programming Language, and First Database System). However, it is pushed into second place as the distance measure appears to be dominated by the numeric fields (categorical fields are given the value 1 if they match and 0 if not), and consequently case 13319, whose numeric function point values are closer to case 10566 than case 10427, is ranked higher even though it has fewer categorical fields in common. This is quite a frequent occurrence – not just in this case but throughout the entire dataset. In many cases this will result in a less appropriate case appearing as the first ranked match which may go some way towards accounting for the relatively poor MMRE values.

Even though the MMRE values are relatively poor for this dataset, they are still below 1. As the value of k increases, then so does the MMRE – quite dramatically – resulting in an average MRE of 6.538 when k = 4. This average is skewed by some extremely high MRE values – as high as 146 in some cases. Case 13319 is an example of this:

- 13319, IFPUG, 47, New Development, other: Sales contact management, ASP, ORACLE, 113, 113

The nearest cases for 13319 are:

- Rank 1, Distance: 0.654

  13700, IFPUG, 352, Enhancement, Process Control, ASP, SQL SERVER, 133, 133

- Rank 2, Distance: 0.663

  10566, IFPUG, 8580, New Development, Financial transaction process/accounting, SQL, Oracle, 359, 359

- Rank 3, Distance: 0.716

  13127, IFPUG, 7496, New Development, Workflow support & management, ASP, SQL Server7, 786, 786

- Rank 4, Distance: 0.730

  10427, IFPUG, 11372, New Development, Financial transaction process/accounting, SQL, ORACLE, 859, 859

Clearly, the effort associated with all these closely ranked cases is some way off the target value (47), but that associated with the second, third, and particularly fourth cases are substantially different. So as k increases the MRE gets significantly larger: 115(refer to footnote[4]) for k = 3 and 146(refer to footnote[5]) when k = 4. Admittedly, this data point is the only one that has a MRE value of more than 100; the rest of the cases result in values less than 8, and the majority of them are less than 1. Nevertheless, this is the main reason that the MMRE is so large. It is a poignant illustration of the impact that outliers, or even the lack of close matches in the dataset, can have on the accuracy of effort predictions. Furthermore, it also demonstrates the rather unpredictable effect of increasing the number of analogies.

5.2. Why does G1-Ran3-33 display such a different trend to G1-Ran2-33?

In contrast to G1-Ran2-33, G1-Ran3-33 has a very different trend of MMRE values, showing a slight downward trend until k = 4 and a very slight increase thereafter. There are no peaks or extreme values as in the case of G1-Ran2-33, and the MMRE values range between 1.716 and 0.909. In some ways this is curious

[4] The mean of the predicted effort is (352+8580+7496)/3 = 5476 and the MRE is Abs(47 – 5476)/47=115

[5] Abs(47 – (352+8580+7496+11372)/4)/47

as the pattern of data in the two sets are not dissimilar as can be seen by the summary table below:

***Table 3**. Summary of Particular Group Dataset*

Dataset	Mean	Median	Min	Max	Skewness
G1-Ran2-33	*2346*	*934*	*24*	*11372*	*1.706*
G1-Ran3-33	*2605*	*1136*	*24*	*11372*	*1.462*

Both have the same minimum and maximum values, so why does G1-Ran3-33 not display any of the extreme values of G1-Ran2-33? From tables 4 and 5 it can be seen that the MRE for the predicted effort based on one analogy is better for G1-Ran2-33 than for G1-Ran3-33. This is caused largely by the poor initial matches for G1-Ran3-33, but also by the frequent predicted effort of 47 for G1-Ran2-33 – often a very poor match but still yielding a MRE value of less than1 (one of the weaknesses of the MRE calculation).

***Table 4:** Top 10 MRE values for G1-Ran2-33 (k=1)*

Case no.	Actual effort	Predicted effort	MRE
13319	*47*	*352*	*6.489362*
15440	*301*	*1136*	*2.774086*
15444	*2504*	*9231*	*2.686502*
14260	*3576*	*11372*	*2.180089*
15720	*934*	*2240*	*1.398287*
10566	*8580*	*47*	*0.994522*
11718	*3934*	*47*	*0.988053*
13896	*1136*	*47*	*0.958627*
13127	*7496*	*352*	*0.953042*
11809	*655*	*47*	*0.928244*

***Table 5:** Top 10 MRE values for G1-Ran3-33 (k=1)*

Case no.	Actual effort	Predicted effort	MRE
13700	*352*	*7496*	*20.29545*
12573	*1671*	*11372*	*5.805506*
10173	*118*	*578*	*3.898305*
10178	*2503*	*11372*	*3.543348*
15940	*66*	*210*	*2.181818*
14260	*3576*	*11372*	*2.180089*
15675	*2762*	*8580*	*2.106445*
14194	*210*	*578*	*1.752381*
13254	*24*	*66*	*1.75*
14485	*484*	*1136*	*1.347107*

In contrast, when four analogies are used the position is reversed and the top MRE values for G1-Ran2-33 are much higher (the value of 146 has already been illustrated) than those for G1-Ran3-33. These values are summarised in the tables 6 and 7 below.

***Table 6:** Top 10 MRE values for G1-Ran2-33 (k=4)*

Case no.	Actual effort	Predicted effort	MRE
13319	*47*	*6950*	*146.8723*
13254	*24*	*540*	*21.5*
11421	*36*	*537*	*13.91667*
12078	*54*	*532*	*8.851852*
13700	*352*	*2231*	*5.338068*
13896	*1136*	*5075*	*3.46743*
15444	*2504*	*7918*	*2.162141*
11132	*278*	*697*	*1.507194*
11100	*2240*	*5346*	*1.386607*
15440	*301*	*691*	*1.295681*

***Table 7**: Top 10 MRE values for G1-Ran3-33 (k=4)*

Case no.	Actual effort	Predicted effort	MRE
13700	*352*	*3388*	*8.625*
12573	*1671*	*7921*	*3.740275*
13254	*24*	*112*	*3.666667*
10178	*2503*	*6911*	*1.761087*
14485	*484*	*1078*	*1.227273*
15675	*2762*	*5709*	*1.06698*
12078	*54*	*104*	*0.925926*
10173	*118*	*227*	*0.923729*
14260	*3576*	*6705*	*0.875*
10802	*578*	*112*	*0.806228*

Although the worst case for G1-Ran3-33 produces a very high MRE value (8.625), this is substantially lower than the value of 146 which is primarily responsible for the overall high MMRE for G1-Ran2-33. Looking at this worst case in more detail it can be seen that the predicted effort values get closer to the actual effort (having started off some considerable distance away), which reduces the MRE. This is in contrast with the case of 13319 in G1-Ran2-33 where the values deviate even further as more analogies are brought into play.

- 13700, IFPUG, 352, Enhancement, Process Control, ASP, SQL SERVER, 133, 133

The nearest data points for 13700 are:

- Rank 1, Distance: 0.693

  13127, IFPUG, 7496, New Development, Workflow Support & Management, ASP, SQL Server7, 786, 786

- Rank 2, Distance: 0.707

  13981, IFPUG, 4648, New Development, Other: Sales Promotion Tool, Visual Basic, SQL SERVER, 895, 895

- Rank 3, Distance: 0.755

  15603, IFPUG, 756, Enhancement, Financial Application Area, Java, Interactive, 124, 124

- Rank 4, Distance: 0.755

  11809, IFPUG, 655, Enhancement, Financial Application Area, Java, Interactive, 113, 113

From this it could be argued that the distribution of projects in the dataset is important: rather obviously, a case base that does not contain projects that are remotely close to those for which predictions are being made is unlikely to produce accurate results. This point is illustrated by group G2-Ran3. The trend for all subcategories in this group is the same: initially disparate values for k=1 quickly converge to a much smaller range as k increases. The MMRE values are still too high for this to be considered a "good" prediction, but the pattern of the graph follows that shape that might intuitively be expected. The reason for this is that the group (and subgroups) consists of data which is spread evenly from the lowest to highest value. All groups have the same maximum (21700) but also contain other large values (19306, 14992 and 11165) which tend to be chosen as close matches to each other and result in relatively good estimates, or at least not very poor ones.

This appears to confirm the observations of Kadoda et al. [6] and Shepperd and Kadoda [15], that there is likely to be a strong interaction between the accuracy of a given prediction system and underlying characteristics of the dataset it is applied to. However, looking at the graphs of the results, it does not appear that increasing the size of the dataset improves the accuracy of the prediction – larger datasets appear to display similarly erratic results to the smaller ones. This interaction between the dataset and the predictions can be clearly observed in the figures 10 to 12 which group the results by different sized datasets.

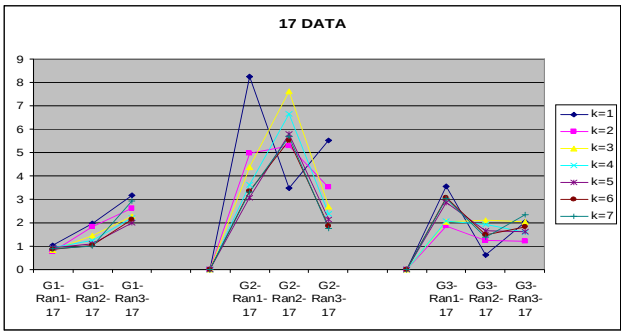

Fig.10. *Result of MMRE vs Groups for 17 data*

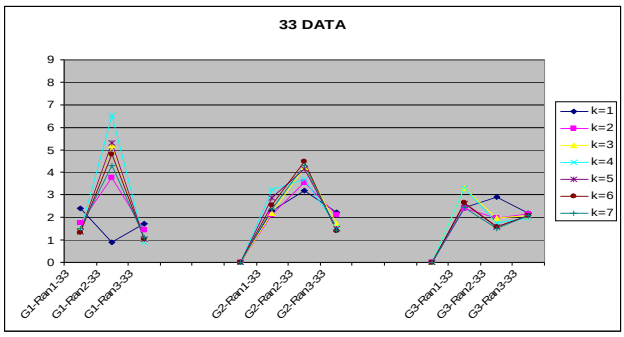

Fig.11. *Result of MMRE vs Groups for 33 data*

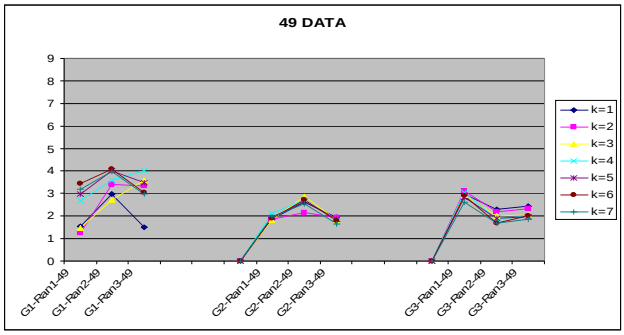

Fig.12. *Result of MMRE vs Groups for 49 data*

G2Ran2-17		k=1	k=2	k=3	k=4
Case no.	**Actual Effort**	**Predicted Effort**	**Predicted Effort**	**Predicted Effort**	**Predicted Effort**
16023	*525*	*1712*	*1556*	*2138*	*1863*
16076	*105*	*51*	*156*	*1341*	*1434*
16612	*465*	*1037*	*2170*	*2380*	*2135*
17461	*1400*	*3303*	*1914*	*1621*	*1644*
17614	*3303*	*1400*	*1218*	*987*	*856*
18030	*2800*	*1009*	*737*	*751*	*823*
18398	*14992*	*11165*	*5973*	*4318*	*3938*
18705	*1009*	*1712*	*1246*	*1764*	*1439*
19107	*1712*	*1009*	*767*	*978*	*1661*
19673	*3712*	*262*	*183*	*693*	*532*
20145	*51*	*105*	*183*	*1359*	*1447*
20426	*147*	*5018*	*10005*	*10391*	*7989*
20896	*5018*	*147*	*7569*	*5979*	*7262*
21180	*781*	*1009*	*1904*	*4991*	*3859*
21550	*11165*	*781*	*895*	*1530*	*1263*
22177	*1037*	*465*	*1884*	*1722*	*1423*
22409	*262*	*3712*	*1908*	*1289*	*1395*

Table 8: *Predicted effort for G2-Ran2-17 (up to k=4)*

5.3. Why is the MMRE for k=1 for G2Ran2-17 so high?

When using only one analogy there is obviously no opportunity to average the results and so the difference in the value of effort could effect the result. In G2Ran2-17 there are two big values in this group (14992 and 11165) and the next value is 5018, followed by 3303 and lower. The presence of these high values could skew the effort predictions. We investigate this further by looking at the result of the data set in both groups (see table 8).

The data points that have the greatest impact on the MMRE are 20426 and 22409, which are considered in more detail below.

- 20426, COSMIC-FFP, 147, New Development, Transaction/Production System, Visual Basic, SQL Server7, 751, 751

The nearest data points for 20426 are:

- Rank 1, Distance: 0.755

  20896, COSMIC-FFP, 5018, New Development, Document management, ASP, SQL SERVER, 762, 762

- Rank 2, Distance: 0.846

  18398, IFPUG, 14992, New Development, Customer Billing/Relationship Management, HTML, ORACLE, 694, 694

- Rank 3, Distance: 0.902

  21550, IFPUG, 11165, New Development, Document mngnt; Financial trans process/acc; Image video or sound processing, Visual Basic, SQL SERVER, 307, 307

- Rank 4, Distance: 0.921

  21180, IFPUG, 781, New Development, Trading, Visual Basic, Oracle 8i, 235, 235

When k=1 for this data point the MRE is 33.13 which is the highest on this group. While when k=2, MRE is 67.06 and the second highest MRE for this group is only 6.28. This again illustrates the impact of the numeric values (the final two size estimates) in the distance calculation. The second data point also illustrates this issue but raises another interesting question:

- 22409, IFPUG, 262, Enhancement, Financial application area, Java, Interactive, 46, 46

The nearest data points for 22409 are:

- Rank 1, Distance: 0.755

  19673, IFPUG, 3712, New Development, Catalogue/register of things or events; Document management; Online analysis and reporting; Workflow support & management, Java, ORACLE, 51, 51

- Rank 2, Distance: 0.756

  16076, IFPUG, 105, Enhancement, Financial application area, Java, Interactive, 19, 19

- Rank 3, Distance: 0.756

  20145, IFPUG, 51, Enhancement, Financial application area, Java, Interactive, 9, 9

- Rank 4, Distance: 0.756

  19107, IFPUG, 1712, Enhancement, Relatively complex application, 4GL, Interactive, 89, 89

Again this leads to similarly high values for the MRE but illustrates another issue with the data. In all cases the size calculations are relatively low numbers of function points (46, 51, 19, 9, 89), but the effort values vary disproportionately (262, 3712, 105, 51, 1712) except where there is a close categorical match where the effort is almost consistently 5.5 times the size. This may be coincidence but may also indicate data which comes from the same company or even the same team. Unfortunately, such information is not available in the data set for reasons of privacy, even though it is potentially useful in finding matching cases.

5.4 Questions arising from the Pred(25) results.

As mentioned at the start the PRED(25) results are not included for reasons of space, even though they are considered a more preferable mechanism to MMRE for assessing the accuracy of prediction mechanisms given the weaknesses associated with MMRE [3]. The PRED(25) results display similar characteristics to the MMRE results: no general trends regarding the accuracy of the estimate and the number of analogies, and a clear indication of the impact of the underlying data set.

6. CONCLUSIONS

The main finding of this study is that no reliable guidance can be given regarding the number of analogies that should be employed in making a prediction. In some cases there is a tendency for the data to converge as k increases whilst in others it diverges. Most of the graphs seem to suggest that the data is having big influence in calculations. The results also do not give any confidence that increasing the size of the dataset results in more accurate predictions. In some cases the smallest set (17 cases) is the least accurate, but in others it is the most! The larger datasets (with 33, 49 and 67 values) tend to gravitate towards each other more and display less volatility, but their relationship to each other is not always predictable.

It was also found that outliers in the form of large or small values could possibly effect the predictions. Related to this is the distribution of data within the dataset – those with a more even spread of data tended to produce lower MMRE values. The quality of the data set seems plays a major role in the precision of the prediction.

Another important result of this study is the relationship between the features used and the distance calculation. In this study only 8 features are employed, and only 2 of these are numeric - Functional Size, Adjusted Functional Points (Effort is also numeric but is

not employed in the distance measure as it is the value which is being predicted) and the rest is categorical. Again the characteristics of the dataset could influence the prediction accuracy because categorical data contributes either 1 or 0 to the distance calculation depending on whether there is a match or not. As a consequence the numeric values appear to dominate the distance calculation resulting in cases which are arguably slightly poorer matches being ranked higher than apparently better ones.

Future work in this area will aim to address these issues, particularly those relating to the spread of data and the distance calculation (and the subsequent adaptation of the analogy) with the aim of making the use of CBR for effort prediction more reliable.

7. ACKNOWLEDGEMENTS

Sukumar Letchmunan would like to gratefully acknowledge the sponsorship of the Science University of Malaysia (USM) and Ministry of Higher Education (MOHE)for supporting his studies.

8. REFERENCES

1. Aamodt, A. and Plaza, E., Case based reasoning: Foundational issues, methodology variations, and system approaches, *AI Communications*, 1994.
2. De Almeida, M.A., Lounis, H., and Melo, W.L., "An Investigation on the Use of Machine Learned Models for Estimating Correction Costs*", Proc. Int'l Conf.Software Eng*., IEEE 1998.
3. Foss, T.; Stensrud, E.; Kitchenham, B.; Myrtveit, I., "A simulation study of the model evaluation criterion MMRE"*, IEEE Transactions on Software Engineering*, vol.29, no.11, pp. 985-995, Nov. 2003
4. Idris, A., Zakrani, A., Elkoutbi, M. and Abran, A. 2008. Fuzzy Radical Basis function Neural Networks for Web Applications Cost Estimation. 4th International Conference on Innovations in Information Technology, 2007. IIT '07. IEEE 2008: 576-580
5. ISBSG Dataset 10(2007), http://www.isbsg.org
6. Kadoda, G., Cartwright, M., Chen, L., and Shepperd, M.J., Experiences Using Case-Based Reasoning to Predict Software Project Effort, *Proceedings of EASE 2000*.
7. Kadoda, G., Cartwright, M., and Shepperd, M.J., Issues on the effective use of CBR Technology for software project prediction. In *Case-Based Reasoning Research and Development,* LNCS, Springer, 2001.
8. Mendes, E., Mosley, N. and Counsell, S., Comparison of Web size measures for predicting web design and authoring effort*. IEEE Proc-Soft Vol 149, No. 3 ,June 2002*
9. Mendes, E., Mosley, N. and Counsell, S. A Replicated Assessment of the Use of Adaptation Rules to improve Web Cost Estimation. *Int. Symposium on Empirical Software Engineering (ISESE '03):* 2003.
10. Mendes, E., Mosley, N., Watson, I., A comparison of Case Based Reasoning Approaches to Web Hypermedia Project Cost Estimation. *Proceedings of the 11th International Conference on World Wide Web,* May 07-11, 2002, Honolulu, Hawaii, USA.
11. Reifer, Donald J. 2000. Web Development: Estimating Quick-to-Market Software. *IEEE Software 2000 17(6)*: 57-64.
12. Ruhe, M., Jeffery, R. and Wieczorek, I. 2003. Cost Estimation for Web Applications. *25th International Conference on Software Engineering (ICSE '03):* 285.
13. Schofield, C. *An empirical investigation into software estimation by analogy*, PhD thesis, Dept. of Computing, Bournemouth Univ., UK, (1998).
14. Shepperd, M. Software project economics: a roadmap. *Future of Software Engineering (FOSE'07)*, 2007.
15. Shepperd, M. and Kadoda, G. "Using Simulation to Evaluate Prediction Techniques," *Seventh International Software Metrics Symposium* (METRICS'01), 2001.
16. Shepperd, M., Schofield, C., Estimating software Project Effort using Analogies, *IEEE Transactions on Software Engineering,* 23(12), pp. 736-743, 1997.
17. Walkerden, F., and R. Jeffery, 1999. An empirical study of analogy-based software effort estimation. *Empirical Software Engineering* (pp. 135-158)
18. Briand,L. C., and I. Wieczorek, 2002. Resource estimation in software engineering. *Encyclopedia of Software Engineering,* John Wiley & Sons, New York
19. Jorgensen, M. "A Review of Studies on Expert Estimation of Software Development Effort.", *Journal of Systems and Software* 70(1-2) : 37-60, 2004
20. Boehm, B. W., E. Horowitz, R. Madachy, D. Reifer, B. K. Clark, B. Steece, A. W. Brown, S. Chulaniand C. Abts . Software cost estimation with COCOMO II. NJ, Prentice Hall, 2000.
21. L. Angelis, I. Stamelos, M. Morisio, "Building A Software Cost Estimation Model Based On Categorical Data," 7th IEEE International Software Metrics Symposium (METRICS'01), 2001
22. S. Conte, H. Dunsmore, and V. Y. Shen, Software Engineering Metrics and Models. Menlo Park, Claif: Benjamin Cummings,1986