Human vs Algorithm

Carolyn Mair
Southampton Solent University, UK
carolyn.mair@solent.ac.uk

Martin Shepperd
Brunel University, UK
martin.shepperd@brunel.ac.uk

Abstract

We consider the roles of algorithm and human and their inter-relationships. As a vehicle for some of our ideas we describe an empirical investigation of software professionals using analogy-based tools and unaided search in order to solve various prediction problems. We conclude that there exist a class of software engineering problems which might be characterised as high value and low frequency where the human-algorithm interaction must be considered carefully if they are to be successfully deployed in industry.

1. Introduction

We have interested in deploying computer science techniques to assist in analogy-based prediction using case-based reasoning (CBR) [3] for more than a decade. Fundamentally this involves searching in p-dimensional feature space for k donor cases (entities of interest) using some distance metric. In addition, we use search heuristics to find some restriction of the feature space in order to optimise the case-retrieval process [2].

In this short position paper we describe some results from an observation study of experienced software project managers where the goal was to investigate the interaction between expert and our CBR tool. To do this we provided each participant with a case-base of 18 completed software projects drawn from their own organisation. Each project comprised an effort value and 15 other descriptive features some continuous and some categoric.

The participants were then asked them to provide a "sanity check" for two further projects. In the first case, unknown to the participants, the true project effort value was divided by three. for the second case the true effort value was left unaltered. In both cases the participants were asked to comment on their level of confidence in the provided 'prediction'. This is an example of a typical project management task where the an independent opinion is sought.

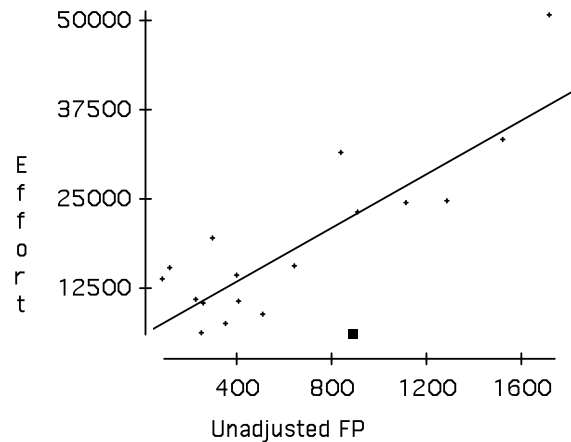The participants were assisted in their use of the CBR

Figure 1. Scatterplot of Function Points (FP) and Project Effort

tool by one of the researchers (MJS) and were also encouraged to use a think-aloud protocol to verbalise their thoughts whilst problem solving. This protocol was recorded and subsequently transcribed.

2. Expert Behaviour

One of the interesting aspects of the prediction task given to our experts was that, as Figure 1 reveals, there is no simple linear relationship between a size measure such as Function Points (FPs) and Effort in person-hours. The highlighted data point indicates an example of an extreme outlier. This meant that in order to successfully retrieve useful donor cases (which did indeed exist within the case base) the participants had to manage multiple dimensions of project feature space. And whilst this is relatively trivial from an algorithmic point of view — our CBR tool uses a modified form of Euclidean distance — it is somewhat challenging for the human.

In general participant P1 did not wish to use the CBR

tool to retrieve similar projects but preferred to establish and validate theories concerning relationships among the data. In particular, P1 sought to find linear relationships and stable ratios, for example in terms of delivery rates yet such relationships are not to be found within this project data set and this caused P1 some difficulty. By contrast, P2 not only decided that the provided estimate was too low (5086 hours) but suggested an alternative value (12000 hours). Recall, the value provided was divided by three so the true value was 15286 hours. At one stage P2 also looked for some linear relationship between FPs and effort but when this was not supported by the data, P1 retreated from this viewpoint and investigated a new approach based on finding analogies using two (not one) dimension. This proved to be successful.

The problem is the design of our algorithms does not facilitate human engagement, consequently the user is forced to treat the searching for analogy process largely as a "black box". Yet project effort prediction is a key part of cost-modelling and therefore important in driving business decisions such as responding to invitations to tender and cost-benefit analysis. As such it may be seen as a high value decision. On the other hand software projects typically have durations of many months and so are relatively infrequent events. In such circumstances it is highly unlikely that experts will be replaced by algorithms. Thus, we need to see the algorithmic approach as a means of augmenting the software engineer.

3. Conclusions

In recent years there has been much interest in the use of search algorithms as an automated problem solving technique for a wide range of applications within software engineering [1]. We merely caution that for important or high-value problems it can be difficult for some experts to simply 'trust' the algorithm. This we showed by our study where one of the participants was unable to meaningfully engage with our CBR tool because he could not properly understand its functioning. One of the causes for this was that the structure and representation are geared for algorithmic ease of implementation rather than the project manager. With the benefit of hindsight this is somewhat unfortunate.

References

[1] J. Clark, J. Dolado, M. Harman, R. Hierons, B. Jones, M. Lumkin, B. Mitchell, S. Mancoridis, K. Rees, M. Roper, and M. Shepperd. Reformulating software engineering as a search problem. *IEE Proceedings - Software*, 150(3):161–175, 2003.

[2] C. Kirsopp, M. Shepperd, and J. Hart. Search heuristics, case-based reasoning and software project effort prediction. In *GECCO 2002: Genetic and Evolutionary Computation Conf.*, New York, 2002. AAAI.

[3] M. Shepperd and C. Schofield. Estimating software project effort using analogies. *IEEE Transactions on Software Engineering*, 23(11):736–743, 1997.