# Towards the Prediction of Development Effort for Web Applications

*Emilia Mendes*
Computer Science Department
The University of Auckland
Auckland, Private Bag 92019, New Zealand
Tel: 0064-9-3737599 extension: 6137
E-mail: emilia@cs.auckland.ac.nz

*Wendy Hall*
Multimedia Research Group
Electronics and Computer Science Department
The University of Southampton
Highfield, S017 1BJ, Southampton, UK
E-mail: wh@ecs.soton.ac.uk

## ABSTRACT

To estimate the effort required to develop Web applications can be quite a difficult task, however accurate estimates of development effort play an important part in the successful management of major Web development projects. This paper describes the use of analogy to estimate the development effort of Web applications. Two datasets were used in the estimation process and the results were optimistic. As the estimation by analogy requires a considerable amount of computation, we have used an automated environment – the ANGEL tool - that supports the collection, storage and identification of the most analogous projects in order to estimate the effort for a new project. We have shown that estimating by analogy is a candidate technique and that with the aid of an automated environment it is a practical technique to apply to Web development.

**KEYWORDS:** Effort estimation, Web development, hypermedia development, estimation, analogy.

## ESTIMATION BY ANALOGY

The rationale for the estimation by analogy method is to characterise the project, for which the estimate is to be made, in relation to a number of variables (attributes). This description is then used to find other similar projects that have already been finished. An estimate for the new project (active project) is made based on the known effort values for the finished projects.

The analogies used in our work are calculated by measuring the Euclidean distance in n-dimensional space where each dimension corresponds to a variable. Values are standardised so that each dimension contributes equal weight to the process of finding analogies [4].

## THE ANGEL SOFTWARE TOOL

We have used an analogy effort estimation tool which automates the process and provides an environment where

data can be stored, analogies found and estimates generated. The tool was developed under the name of ANaloGy softwarE tooL (ANGEL) [4]. One of the important features of ANGEL is the ability to determine the optimum combination of variables for predicting effort.

Once we have chosen the variables to use in the estimation the next step is to run the estimation algorithm so that we can obtain not only the estimation suggested, but also the level of confidence that we can have in using analogies drawn from a given dataset.

## THE EMPIRICAL RESULTS

The two datasets used were obtained from a case-study evaluation where 45 second-year Computer Science students were asked to develop Web applications aimed at teaching Human Computer Interaction concepts. Six data points were discarded since they contained incomplete information. One of the datasets has the data from students whose authoring experience ranges from 0 (no experience) to 5 (on a scale from 0 to 10) - LEL group - and the other dataset represents the students whose authoring experience ranges from 6 to 10 (a great deal of experience) - HEL group. The case study is described in detail in [3].

The variables collected were as follows:
- The number of documents, which equates to the number of HTML files that the application has. The Web applications considered are 'static', since the number of dynamically generated pages is quite small, if not absent.
- The number of reused documents, which represents the number of documents that have been reused from elsewhere.
- The number of links that the Web application has. The links considered here are 'static', i.e., not dynamically computed.
- The compactness [1] of the application, which indicates how inter-connected the documents are. A completely connected application means that, from each document, there are links reaching all the other documents in the application. Compactness was measured in the case study evaluation by asking subjects to give an estimate of it, from 1 (completely disconnected) to 5 (completely connected). Consequently, the compactness measured was the perceived compactness.
- The Stratum [1] of the application, which indicates to what degree the Web application is organised for

directed reading. A sequential application represents an application where there is only one path to be followed by the reader. For the case study evaluation, subjects were asked to estimate the stratum of the application, from 1 (no sequential navigation) to 5 (sequential navigation). Consequently, the stratum measured was the perceived stratum.

- The structure of the application - sequence, hierarchy, or network, which represents the way in which the documents were linked. A sequential structure corresponds to documents linearly linked; a hierarchical structure denotes documents linked in a tree shape and a network structure stands for documents linked in a web shape.

The HEL group contains information about 17 projects whereas the LEL group contains information for 22 projects. The variables selected for the datasets that we used were imposed by the fact that the data had already been collected. However, the datasets had at least four variables which were size related.

In order to evaluate how precise the estimations would be for each one of our datasets, we followed the subsequent steps:
- One project at a time had its development effort estimated.
- Using ANGEL we calculated the best combination of variables to be used as effort predictors [4].
- The effort was then predicted using 1, 2 and 3 analogous projects [4].

For the LEL group our conclusions are as follows:
- The mean for number of hours predicted using 1 analogy was very close to the mean for number of hours measured.
- The variables which contributed the most for the estimations were number of links and structure (12 projects), followed by number of links, structure and number of reused documents (7 projects).
- For some projects the estimated effort was nearly the same or even the same as the actual effort.

For the HEL group our conclusions are as follows:
- The mean for number of hours predicted using 3 analogies was the closest to the mean for number of hours measured.
- The variables which contributed the most for the estimations were number of documents reused (5 projects), followed by number of documents, compactness, stratum, number of reused documents and structure (4 projects).
- Only four projects had estimated efforts similar to the actual effort, suggesting that in order to obtain the best results the datasets used need to be as homogeneous as possible.

Even though the datasets used in our work were not collected with analogy in mind, and therefore the variables are not particularly well suited to characterising projects, we have obtained some positive results, which suggest that estimation by analogy could be a viable option for some types of datasets.

## CONCLUSIONS AND FUTURE WORK

The two datasets used in our work had four variables that characterised the applications' size in some way. However, as size can be described in terms of length, functionality and complexity [2] further investigation is necessary in order to identify the best combination of variables suitable to give the best results when applying estimation by analogy.

Further work includes the comparison between estimation by analogy and two algorithmic models, namely linear regression and stepwise multiple regression. Although literature in software estimation has showed that the analogy method outperforms or equals the best algorithmic method [4] we want to investigate whether or not the results would be similar when applied to the development of Web applications.

To conclude, there is an urgent need for adequate Web development effort prediction at an early stage in the development. As the use of the World Wide Web as a resource delivery environment increases, effort estimation can contribute significantly to the reduction of costs and time involved in developing Web applications. In this paper we have shown that analogy may be a viable estimation method for prediction, particularly when given the necessary tool support.

We do not, however, wish to create the impression that analogy based prediction should replace algorithmic methods. Dissimilar datasets will have different characteristics implying that a variety of techniques should be contemplated.

## REFERENCES

1. Botafogo, R. A., Rivlin, E., and Shneiderman, B. Structural Analysis of Hypertexts: Identifying Hierarchies and Useful Metrics. ACM TOIS, 10, 2, (1992), pp. 143-179.

2. Fenton, N. E., and Pfleeger, S. L. Software Metrics, A Rigorous & Practical Approach, 2nd edition, PWS Publishing Company and International Thomson Computer Press, (1997).

3. Mendes, M. Emilia X., Hall, Wendy and Harrison, Rachel Applying Measurement Principles to Improve Hypermedia Authoring, to be published at the New Review of Hypermedia and Multimedia, (1999).

4. Schofield, C. An empirical investigation into software estimation by analogy, Unpublished Doctoral Thesis, Dept. of Computing, Bournemouth Univ., UK, (1998).