Effort Estimation in Information Systems Projects using Data Mining Techniques

JOAQUÍN VILLANUEVA-BALSERA
FRANCISCO ORTEGA-FERNANDEZ
VICENTE RODRÍGUEZ-MONTEQUÍN
RAMIRO CONCEPCIÓN-SUÁREZ
Project Engineering Area
University of Oviedo
ETSIMO, C/ Independencia 13 Oviedo (Asturias) 33004
ESPAÑA
(fran@api.uniovi.es)

Abstract: - Project scheduling is a crucial task that can lead to project failure due to the lengthening of its duration or a wrong estimation of the needed effort to implement it. It is necessary to have available a tool that help us learn more about the project in order to choose the influential attributes on the project deviations and also to provide us with more accurate estimations.
This paper analyses the feasibility and the advantages of the development of a system based on Artificial Intelligence techniques capable of selecting the attributes that affect the project duration and the effort it requires to make it possible through a data set of software projects historical information, against current techniques.
To do that, it is proposed a method for the analysis of existing data set and their pre-processing, to obtain a model that can meet the project manager standards.

Key-Words: Effort estimation, data mining, project management, scheduling

1. Introduction

One of the phases in the project management is the effort and duration estimation of each of the activities it consists of. Therefore, it is necessary to have proper information on the duration and effort by person/month needed to implement each task.

Having this information will make project management easy both at cost and schedule levels, making the distributions of resources easier and diminishing the risks or critic zones.

The historical information, derived from the closure of former projects [3], will provide the knowledge basis on which to apply the techniques to extrapolate this knowledge and use them on future projects.

The effort control and accuracy through the project life cycle and the compilation of the characteristic data of each project when this comes to an end produce a continuum correction effect which allows fixing specific models which will help the planning of new projects as a whole: effort, cost, schedule and resources.

It is necessary to have a tool that helps us learn more about the problem and allow us to select the influential variables on the project deviations and to provide us with more accurate estimations.

This paper analyses the development of a system based on artificial intelligence techniques that is capable of selecting the variables that affect the project duration and the effort it requires to make it possible through a set of historical data.

To develop the system, we used a data set belonging to the International Software Benchmarking Standards Group [1], which was compiled based on information extracted from more than 2000 projects. These data contain numerical and categorical values, with a great number of values missing. It is for this reason that the values were thoroughly pre-processed and artificial intelligence techniques which best fitted data conditions were selected.

In the development of this paper, the scope of the problem to solve is described first, followed by a summary of the most used current techniques. Given that we start from a historical data set taken from former projects, a data mining methodology will be applied. The CRISP DM [2] methodology, used to solve this problem, will be described.

Finally, it will also be described the modelling method and the conclusions we have drawn to solve the problem faced.

2. Problem description

The estimation of the number of people needed for an information systems project consists of the implementation of a series of techniques and methods which a company uses in order to learn beforehand the cost that the system analysis, development, implementation and tests will entail. The accurate estimation of the time and resources necessary to the development of a project, which is essential to the perfect development of any project, especially in the computer field, where the budgets and schedules are often exceeded, leading to the project failure.

The accurate prediction of the necessities of an information system project is a critical activity when it comes the time of taking managing decisions and determining in detail the effort and the dedication the project manager, analysts and programmers should apply. Without a reasonable capacity of effort estimation, the project managers will not be able to determine how much time and how many resources the project requires which means this one is out of control since the beginning. The analysts will not be able to make the right analyses during the design stages and the project staff will not be able to tell their clients and managers that their schedules and budgets are unreal. This could lead to false optimism and the unavoidable delays and deviations.

Despite talking about the term “cost estimation”, the output values in information system projects are not usually measured in terms of currencies. The estimations are often assessments of the expected effort for the development of the project and the time schedules required to accomplish it.

This is a non-physical existence product whose main cost relies on its development or design (nor on its making or its replication from the first copy), therefore we find it logical to assume that its production cost are mainly based on personnel expenses, using the per person/month or per person/year measure.

There are also other reasons that make the project estimation difficult, company pressures being among them (to diminish the cost or time needed) and the fact that there is a generalized lack of data on concluded projects (software size, cost, productivity, etc.) which could help professionals when making estimations.

All current methods depend on the amount of available information. The more the project makes progress, the more details and accurate information would appear, therefore the estimation accuracy gradually improves. For this reason, the estimation must always be a continuous process, with constant refinements and improvements, more than a specific activity.

3. Historical approach to the problem

To collect the necessary data to solve the problem faced, it is first studied other methods currently used to learn which the influential attributes are and to help defining the problem.

The cost of the software development is a function basically for the personnel needed and this derives from several factors related to the project, human resources, development conditions and the final product. The product identification is made through measurements that characterize its size, which is the primary factor in all estimation models.

There are two usual ways of measuring the size of an information systems project: the Code Lines and the Function Points.

The Function Points [4] Analysis is a measurement that quantifies the functionality to be delivered to the users when building a program. The first proposal of Function Points made by A. J. Allbrecht has undergone several refinements and different versions have appeared since. All different varieties of functional points lie on data which imply the existence of a more or less formalized specification.

The originality of this method is that it allows us to measure the size of information system projects from the point of view final users have about the required functions of the program, not worrying about technologies, tools or programming languages that will be used.

The function points classify these views into five types of functionality:

- Inputs: to this category belong all communication contributions of the program users.
- Outputs: all communication contributions of the program with the user.
- Internal logic files: main logic files from the users point of view.
- Interface files: files to interact with other programs.
- Queries: all inputs that must cause an immediate output.

Once these factors have been considered depending on the real result of functions, we obtain the non-fixed function points. To know the real effect of the functions, it is necessary to incorporate other data which can introduce those factors affecting the program as a whole. These fixing factors take into account circumstances such as human or technological factors.

Another method is the MARK II, an evolution of the Allan J. Albrecht model, being its main characteristic the consideration of the system as a collection of logic transactions composed by input, process and output components.

Once the functions points are fixed, to calculate the size of the project, multiply the calculated value by the size each function point is valued. This evaluation could be

different depending on the organization.
Other methods to be used to estimate the information system project effort is the "expert's knowledge", which is asking for the advice of professionals in that particular field.
Another technique is the estimation by analogy, which consists of a more formal version of the expert's knowledge, where the project to be developed is compared to one or more closed programs with available data. Depending on the similarities and differences with those projects, it infers the cost of the new development.
Another set of methods are the so called "empirical estimation models". They are, as a whole, mathematical formulae which relate the different parameters of the project (size of the software, project environment conditions, etc.) to the effort required. SLIM and COCOMO are among them.
COCOMO [5] is an empirical estimation method and it is based on data coming from experience. It consists of estimating the effort per person/month based on the size measured in code lines and also the duration of the project based on the effort. It also uses fixing parameters according to the type of method of project development, which can be "organic", "semidetached" and "embedded".
From these basic equations, COCOMO differences three different models which correspond to the different information quantities available in the different stages of the life cycle, which are basic, intermediate or advanced.
These models consider the effort E as result of an equation based on:

$$E = a \cdot S^{b} \quad (1)$$

Where E is the effort, S is the project size in code lines, a reflects the productivity and b is a scale economy factor.
To make the final calculations of the necessary effort, separately form the formulae, an effort adjusting factor must be applied, including product, hardware, personnel and project attributes.
Afterwards, another evolution comes, COCOMO II [6], which heads to the following three different phases of the spiral life cycle, which are: programs development, anticipated design and post-architecture. Also, the three exponent modes have been changed and replaced by five scale factors.
On ISBNG dataset, previously commented in the introduction, different studies have been made as it is the case of [7] that makes effort estimation by analogy using information from previous similar projects to predict the effort for a new project.

4. Methodology

Mining techniques can help in data analysis, modelling and optimization. The software estimation process is influenced by a lot of variables. In order to get a successful model a work methodology must be use for Data Mining projects. CRISP-DM [2] is one of the most usual process models. It divides life cycle for Data Mining projects in six phases.
The methodology CRISP-DM constructs the cycle of life of a project of data mining in six phases, which interact between them on iterative form during the development of the project.

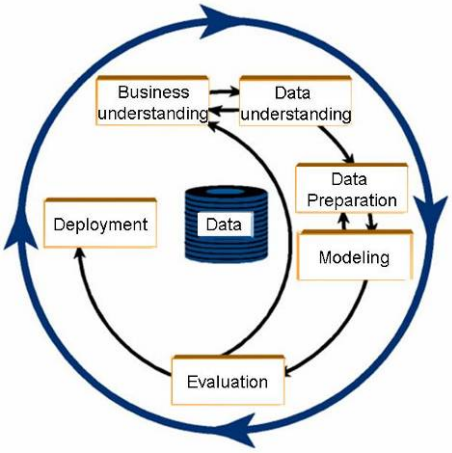

Figure 1. Phases of the Modelling process of methodology CRISP-DM.

The first phase, business understanding is an analysis of the problem, includes the understanding of the objectives and requirements of the project from a managerial perspective, in order to turn them into technical objectives and plans.
The second phase, data understanding is an analysis of data includes the initial compilation of information, to establish the first contact with the problem, identifying the quality of the information and establishing the most evident relations that allow establishing the first hypotheses.
Once, realized the analysis of information, the methodology establishes that one proceeds to the data preparation, in such a way that they could be treated by the modelling technologies. The preparation of information includes the general tasks of data selection to which the modelling technology is going to be applied (variables and samples), data cleanliness, generation of additional variables, integration of different data origins and format changes.
The phase of data preparation, it is more related to the modelling phase, since depending on the modelling technology that is going to be used, the data need to be processed in different forms. Therefore the phases of preparation and modelling interact between then.
In the modelling phase the technologies more adapted for the specific project of data mining are selected. Before proceeding to data modelling, it must establish a design of the evaluation method of the models, which

allows establishing the confidence degree of the models. Once realized these generic tasks one proceeds to the generation and evaluation of the model. The parameters used in the generation of the model depend on the data characteristics.

In the evaluation phase, the model is evaluated in that degree they are fulfilled of the success criteria of the problem. If the generated model is valid depending on the success criteria established in the first phase, one proceeds to the development of the model.

Normally the projects of data mining do not end in the model implantation but it is necessary to document and present the results of an understandable way to achieve an increase of the knowledge. In addition, in the development phase it is necessary to assure the maintenance of the application and the possible diffusion of the results [8].

5. Modelling method

Following the steps given by the methodology, the collect initial data is made for its later preparation and modelling.

In order to begin the data understanding, which comes from the historical data base that has provided ISBSG [1], it contains a repository of more than 2.000 projects. These dataset inform us about:

- Size metrics
- Efforts
- Data quality
- Type and quality of the product: information relative to the development, the platform, the language, the type of application, organization, number of defects, etc.
- CASE tools utilization
- Team size and characteristics
- Schedule information
- Effort ratios and Function points
- Type of project, product or equipment.

Next it comes to make a data exploration, being detected a great amount of missing values and categorical variables. This situation requires the accomplishment of an important effort for the data pre-processing, to adapt it to the artificial intelligence techniques.

In the following figure, it is shown the percentage of variables of the data base with missing values. Therefore, 70% of the variables have more than 10% of missing values, worsening the problem for some variables with more than 90% of the missing data.

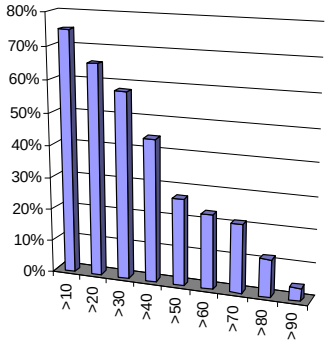

Figure 2. Percentage of variables with missing values.

In this paper, it has not been used traditional techniques of missing values processing, such as deleting the record, filling up the empty data with constants, averages or possible values using distances, since we do not have enough information about the meaning of the information absence. For the missing data processing, it has been chosen to select a technique that allows the handling of this data type, as networks SOM [9] and MARS [10].

Different techniques for the transformation and pre-processing of these variables categorical have been considered. When the existing number of classes were reduced (inferior to six), the action given has been to create as many variables as classes. Thus, for instance, if the categorical variable "*development platform*", contained the values *MR*, *MF* and *PC*, three codified variables have been created such as (1,0,0) if the value of the variable is *MR*, (0,1,0) if *MF* and (0,0,1) if *PC*.

When the number of classes of a variable is very high (superior to six), it has been chosen to directly transform the value of the category to a numerical value.

It has been chosen to apply SOM technique to classify the input data and to find relations between the variables, as well as projections of the input space to detect outliers or clusters.

Once the data have been pre-processed, a SOM map has been made, as it is possible to see in Figure X, to analyze the behaviours and the relations among variables. It can be observed that the variables labelled *WE_X*, which refer to the effort in hours at different phases of the project, are similar in such a way that the projects with high or low efforts (red and blue colour respectively) are grouped in the same zones of the maps. Nevertheless, one detects that projects that have required a high effort in the initial phase, do not accuse that effort in the consecutive phases.

In ISBSG dataset, there is some target variables to be selected as output variable because they measure the effort with different criteria. After making an analysis of the same ones it has been chosen to select one in which the effort is measured in hours by function points (labelled *H_PF* in Figure 3). Map SOM detects

relations among the target variable and the Function Points, the maximum size of the equipment and the total time until the end of the project. Nevertheless, it is observed that the "*lines of code*" is not a variable that is much related to the target variable.

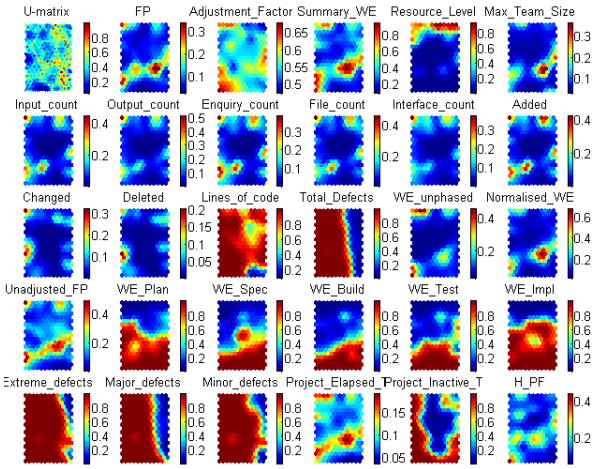

Figure 3. SOM map (Self-Organizing Maps) with variable candidate.

6. Results

From the results obtained in the previous phases, a model for effort estimation was made next. For the creation of the model, different possibilities were analyzed, selecting finally MARS technique by their robustness for the management of the variables with missing values and categorical data, as well as by its capacity to extract relations among variables and to determine the most significant variables.

In order to guarantee that the model is able to generalize the results, the data have been divided into three separated random sets: one of them containing 75% of the data and have been destined to the model training, 10% for the model test and selection of the best model.

The results have been validated with 15% of remaining data.

For the construction of MARS model it has been used the following parameters, interrelation between variables at level 3, base functions of second degree.

The results are in the following table.

Absolute erro	% Old	% Train success	% Validation success
1	22%	25%	16%
5	30%	45%	37%
10	33%	78%	72%
15	41%	100%	96%
20	44%	100%	99%
25	48%	100%	99%

Table 1. Model results.

This is a significant improvement with respect to the reference old model that is a model based on analogies.

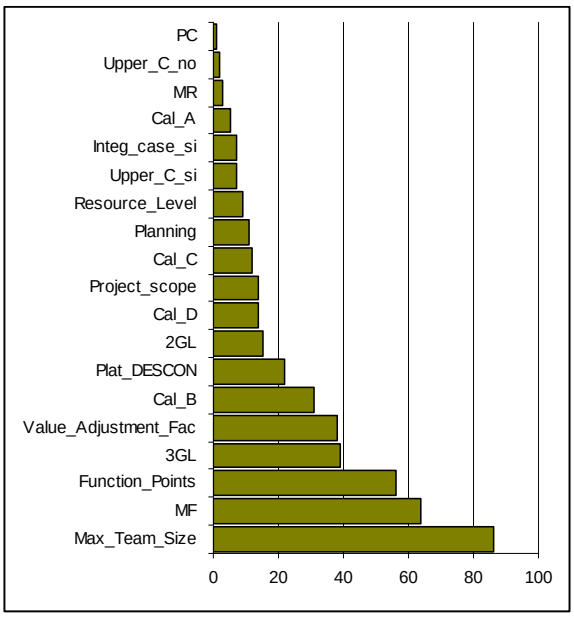

Figure 4. Importance of the variables by MARS.

Once generated the model, a sensitivity analysis of the relative importance of the variables is made, detecting that the attribute that more information contributes to the effort estimation is the *equipment maximum size*. Also they have been detected as important, as it was possible to expect, the measurement of the *function points* and the *adjustment factor value*.

Another important attribute is the *used development platform* and the *type of language* that is used in the programming. Nevertheless, these variables present a great amount of missing values. The absence of information in these two variables has a great relative importance for the effort estimation to the model.

Also an informative parameter with respect to the quality of the information has been introduced in the model, in this case it has been divided into four categories that are: convincing, correct, not proven and little credibility.

The model also considers if a *code upgrade* has been made, if *planning* has been used, as well as other variables related to the *metric* used and the *implication of the resources* in the project.

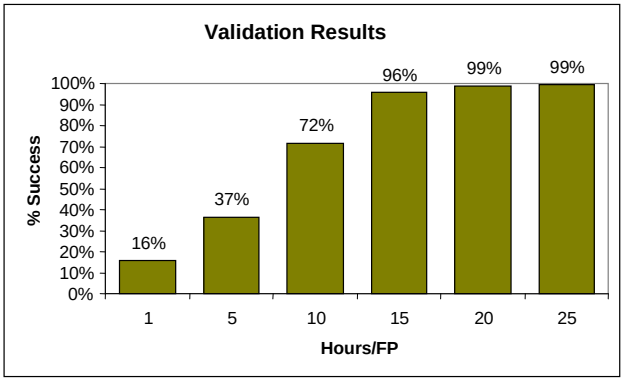

Figure 5. Results obtained by the model

In the model results figure, it is possible to see that with an error of 15 hours by function point, that represents a margin of 5% of relative error, a success of 96% is obtained.
In order to verify the results of the model obtained with techniques of artificial intelligence, it is compared with a model that uses the knowledge of the ISBSG dataset based on analogy techniques, obtaining an improvement in the estimation of a 10%.

7. Conclusions

All the norms or methodologies of projects management that exist at the moment emphasize the importance of the management of time and costs within any type of project and in the projects of information systems due to their own peculiarities.
The chosen system to make the effort estimation has to have the confidence of the project manager and to allow adapting to the changing necessities of the production of the new information systems.
The historical data compiling in the closing of the project is essential to update the data base of projects, so that the system can fit its parameters to the changing conditions of the information systems.
It has been verified that the techniques of data mining are very appropriate for the dataset analysis as changing as the effort estimation, costs or time in information systems, which need a particular adaptation for each organization, as well as they adapt to a set with these characteristics of great volume of missing data and great amount of categorical variables.
This model could be customized for another type of engineering in which it is necessary to consider the effort, and that has tasks in which the factor that it generates the risk is the human resource, as it is the case of the computer science projects.

References

[1] ISBSG, International Software Benchmarking Standards Group. http://www.isbsg.org/.

[2] CRISP DM, Cross-Industry Standard Process for Data Mining. http://www.crisp-dm.org/.

[3] ISO 10006. Guide of quality management. Quality in Management of Projects.

[4] IFPUG, International Function Point Users' Group. http://www.ifpug.org/.

[5] Boehm, B. W., *Software Engineering Economics*, Prentice Hall, Englewood Cliff NJ, 1981.

[6] Boehm, B. W., Clark B., Horowitz, E. Et al., "Cost models for future life cycle processes: COCOMO 2.0", *Annals of Software Engineering* 1 (1), 1-24, 1995.

[7] Jingzhou Li & Guenther Ruhe & Ahmed Al-Emran & Michael M. Richter, "A flexible method for software effort estimation by analogy", *International Symposium on Empirical Software Engineering*, ACM/IEEE, 2006.

[8] Boehm Rodríguez M. T., Ortega F., Rendueles J. L., Menendez C., "Combination of Multivariate Adaptive Techniques and neural networks prediction and control of internal cleanliness in Steel Strips", *Proceedings of EUNITE 2003*. Oulu 2003.

[9] Kohonen T. "Self-Organizing Maps". *Springer Series in Information Sciences*, Vol. 30, Springer, Berlin, Heidelberg, New York, 1995

[10] Friedman, Jerome H. "Multivariate Adaptive Regression Splines". *The annals of statistics*. Volumen:19, Nº 1, pag. 1-141, 1991.