A Solution to Improve the Effort Estimation Error in Software Projects

Gabriela Robiolo and Juan M. Ale

Departamento de Informática - Universidad Austral
Av. Juan de Garay 125 (C1063ABB) 54-11-5921-8000 - Bs.As - Argentina

{grobiolo, jale}@austral.edu.ar

Abstract. *Nowadays users are not highly satisfied with the results obtained when using function points (FP) and other metrics and techniques in order to estimate effort: the estimation errors are still too big. The solution hereby presented is the use of a set of three metrics, Transactions (T), Entity Objects (EO) and Paths (P), as variables of the multi-relational model, when employing the neural network (NN) technique. A case study demonstrates that the estimation error could be thus reduced. It should be pointed out that it was possible to apply the Data Mining technique because the hours worked on each use case had been duly registered. This level of detail about the data has derived into a higher level of precision.*

Resumo. *Hoje em dia, os usuários não estão muito satisfeitos com os resultados obtidos com os pontos de função (PF) e outras métricas e técnicas para estimativa de esforço: os erros de estimação são ainda demasiado grande. A solução aqui apresentada é a utilização de um conjunto de três indicadores, transações (T), Entity Objects (EO) e Caminhos (P), como variáveis do modelo multi-relacional, quando se emprega a rede neural (RN) técnica. Um estudo de caso demonstra que o erro de estimativa pode ser assim reduzido. Deve-se salientar que foi possível aplicar a técnica de Data Mining, pois as horas trabalhadas em cada caso de uso tenha sido devidamente registrado. Este nível de detalhe sobre os dados obtidos em um nível mais elevado de precisão.*

1. Introduction

There were very many papers written on effort estimation metrics and techniques in the last two decades, which shows that this subject has been an important concern for the software industry. At the same time, the error estimation reported in research articles does not satisfy the industry necessities. For an estimator, it is very important to know what the error interval he/she is working with, and the accuracy with which he/she is estimating. This aspect becomes more important if the estimation is being made for a big project, as 1% or 2% differences in the error estimation may mean a big amount of money.

A research work made in 2009 showed alarming error estimation results, as reported in Robiolo (2009). It analyzed sixteen articles written by different authors who had used different estimation methods and had obtained different values of MRE (Magnitude Relative Error). The range of variability of MRE reported was very wide and worrying [Mean 0.37, Standard Deviation 0.7, Range values 0-6.94]. If the quality criterion were to be set at a MRE no bigger than 0.25, then only 4 articles out of 16 may be considered relevant.

It also showed the current dispersion in both the technologies used to estimate, such as Bayesian Network (BN), Stepwise Regression (SR), Case-based Reasoning (CBR), early Web size measures, CART, COCOMO, COSEEKMO, Analogy based method, linear regression (LR) -both simple and multiple- historical productivity, experts' estimation, Vector Prediction (VP), as well as the measures used (several variables of projects, web sizes, Function Points (FP), Use Cases (UC), Use Case Point (UCP), Use Case Rouge (UCR), lines of code LC, Use Case Size Points (USP) and Fuzzy Use Case Size Points (FUSP)). This explains the variety of results obtained.

It is convenient to highlight that the smallest value for the MRE was obtained by the following methods of prediction:

a. Case Based Reasoning and Stepwise Regression, for two sets of Web applications (15 and 37 projects, respectively). The reported MRE was [0.02 - 0.32].

b. Vector Prediction for a set of eight industry projects of different domains, organizations and application types. The reported MRE was 0.20.

The compilation included articles which were based on big historical data bases, which were multi-company or specific-company datasets; medium size datasets; small historical datasets and also one project estimation which was developed by applying four different development processes. Only two of those articles included academic projects. The wide-ranged results may have been caused by two different factors: the metrics used or the technology applied. Even though several techniques were used, the survey shows that this fact did not bring about any significant improvement. **This points out the necessity for a new approach to measure the size and complexity of applications**.

Fenton and Pfleeger (1997) suggested that the unit of measurement to be used had to be easy to understand, simple and consistent with the theory of measurement. In an article where she analyzed the problem with function points, Kitchenham (1997) concluded that:

a. Such measure should be a simplification of the FP concept, in such a way that a basic count could be made in an automatic form from an early representation of the system.

b. It should not add weight to quantify the different degrees of complexity.

c. It should not be an aggregation of different elements.

In order to try to reduce the effort estimation error, we decided to design a case study, for which a set of three metrics was defined and a non parametric technique was chosen. The metrics selected were Transactions (T), Entity Objects (EO) and Paths (P) because they comply with the following conditions: they are based on use cases; the metrics are simple; they may be used at an early stage of the software development process; they reduce the estimation error; they solve the following use case limitations: granularity and level of detail; they are understandable for the project stakeholders; and they help them track the project. The groundings why these metrics were proposed and the conditions in which they were used are fully described in [Robiolo 2008]. A deep analysis of related work where it was justified the use of the metrics is in [Robiolo

2009]. Moreover, the metrics also satisfied the characteristics suggested by Fenton and Pfleeger (1997), and Kitchenham (1997).

Once the metrics and the technique had been chosen, the projects that would be used in the case study were selected. Given the fact that the available information was presented in the form of tables that synthesized, on the one hand, information at project level, and on the other hand, information at use case level, a multi-relational model became a need [Dzeroski 2003]. This model would let us represent the problem as a two-interrelated-table model, in which the projects would be entities and the use-cases related entities.

A tool that would support this type of model was searched for among the set of traditional data mining tools. The one we chose was the SQL server (the 2008 version was used). When running it, we found out that the data was scarce to be used with the majority of the data mining techniques, except for neural network (NN), which could be run in spite of this limitation.

At this point, two research questions arose:

a. When using the metrics T, EO and P as variables, does the technique NN applied to a Multi-relational model improve the effort estimation error, if it is compared to the results reported by other authors?

b. Given the fact that registering the hours worked on every use case is not common practice in the software industry, why does this seem to be a need now?

This article was written to clarify these aspects. The next section describes the model–variables. Later on, the case study will be described and the final conclusions presented.

2. Model Variables

In the following sections, T, EO and P -metrics proposed to improve the estimation error-, Function Points (FP), and other variables included in the multi-relational model are described.

2.1 Measures Used

T, P, and EO were defined as simple units of measurement which, respectively, capture three essential attributes -function, complexity and data - based on information usually available from use case descriptions [Robiolo and Orosco 2008 and Robiolo et al. 2009].

Transactions count the number of stimuli originated by the users and addressed to the system. In Fig. 1 a use case is shown. The two verbs that define the T of the use case are marked in italics. The two stimuli that the actor (user) performs on the system are identified by the followings verbs: choosing and saving a new product. Table 1 shows the verbs –actor's actions-, whether they are taken into account when the counting is made, and the reasons why this happens.

It must be noted that the name of the metric is a bit misleading for people used to FPA terminology: Robiolo's transactions are at finer granularity level than FP transactions.

1. The system displays a screen with the list of **categories** and **subcategories**.

2. The *user* must *choose* the category to which the new **product** belongs. The list of properties of the selected categories and subcategories will appear below. IF THE categories and subcategories are new, a detailed description is also displayed.

3. By pressing enter, the user will go to a new screen in which he should fill in gaps with the definition of the properties of the product.

4. The use case finishes once the *user* enters a product, *saves* the changes, and leaves the system or cancels the operation.

Fig. 1. Description of the use case "Enter a new product into the catalogue"

Table 1. The actor and verbs of the use case "enter a new" product into the catalogue

Verb	Count	Justification
Choose	√	In this example, it is a stimulus to the system, because it triggers a query to the data base. If it had only been a choice made out of a limited number of possibilities, which does not involve consulting the database, it would not have been counted as a stimulus
Press	×	This action is part of the above mentioned action of choosing.
fill in	×	This is the step that takes place before the stimulus on the system is performed
Enter	×	This is part of saving
Save	√	It is a stimulus to the system

Path is a measure of complexity of use cases, which is based on applying the principles of McCabe's complexity measure [McCabe 1976] to the descriptions of use cases, in terms of scenarios. In fact, use cases are usually described by giving a main scenario, which accounts for the 'usual' behavior of the user and system, and a set of alternative scenarios, which accounts for all the possible deviations from the normal behavior, which should be supported by the system. Robiolo (2009) applied to the use case textual descriptions the measure applied by McCabe to code. **Every different path in a given use case scenario contributes to the measurement of the use case's complexity.** For example, in Figure 1 the T identified by the verb "choose" has an alternative path identified by the "if the" expression. So, for each use case transaction, the number of paths is the number of scenarios (main and alternative) [Lavazza and Robiolo 2010].

Entity Objects are the persistent data included in each use case. In the use case shown in Figure 1, the three different entity objects that compose the use case are pointed out in bold letters.

2.2 FP

Albrecht introduced the concept of function points in 1979 [Albrecht 1979]. As FP are defined from the customer's point of view –which does not depend on technology- they take into account elements that can be identified by an external user. Function Points define five basic elements: External Input (EI), External Output (EO), External Inquiry (EQ), Internal Logical File (ILF) and External Interface File (EIF). The complexity of each element is classified in Low, Average or High, and for each level of complexity a factor of complexity is assigned. Therefore, Unadjusted Function Points (UFP) are defined as the sum of the products of the factors of complexity (CF_{ij}) by the number of classified elements (E_{ij}) in an application. This can be expressed as follows:

$$UFP = \sum_{i=1}^{5}\sum_{j=1}^{3} CFij * Eij\,.$$

2.3 Other variables

Also, two variables that characterized the projects were included:

a. CONTEXT, which identifies the context in which they were developed, whose possible values are industry or academic

b. WEB, which states if it is a web application or not.

3. CASE STUDY

This case study was made up of eleven small business projects which were developed in two different contexts: an advanced undergraduate academic environment at Universidad Austral and the System and Technology (S&T) Department at Universidad Austral. The involved human resources shared a similar profile because those who worked at the S&T Department were also advanced undergraduate students. The projects were selected because they met these requisites:

a. The definitions of requirements were based on use cases.

b. There was available information about the hours worked.

c. They were new developments.

d. The use cases were completely implemented.

The characteristics of such projects are shown in Table 2. P1 to P7 come from an academic context and P8 to P11 come from the S&T Department.

Table 2. Characteristics of the projects

Project	Application description	Implementation enviroment
P1	Controls trips using mobile phones	Java, Tomcat, J2ME Mobile Phone Emulator
P2	Manages a centralized web purchase system	Java, Tomcat, Hibernate
P3	Manages a hotel information system	Java, Struts, Hibernate
P4	Tracks projects information	Jave, Hibernate, Tomcat
P5	Manages information about music bands	Java, Apache, Tomcat
P6	Sells with an authorized card	MS (Visual Studio, Office Sharepoint Portal Server)
P7	Sells cinema tickets using mobile phones	MS Windows XP Professional, Apache, Tomcat
P8	Tracks files	JSP, J2EE, Hibernate, Tomcat
P9	Manages an accounting process	JSP, J2EE, Hibernate
P10	Manages university graduate alumni information	Power Builder, Javascript
P11	Controls clients' debts	Power Builder

Table 3. Data of the projects

Project	AE [PH]	T	EO	P	UFP
P1	410.00	57	18	71	144
P2	473.50	63	21	73	266
P3	382.40	48	22	60	171
P4	285.00	35	8	49	142
P5	328.00	21	5	34	89
P6	198.00	23	11	35	75
P7	442.02	33	10	50	57
P8	722.65	79	17	97	210
P9	392.00	56	19	83	311
P10	272.00	39	16	42	155
P11	131.00	18	9	18	117

The available data was displayed in the form of two tables. The first one –Table 3- synthesized information about the variables Actual Effort (AE) -measured in person hours-, T, EO, P, and UFP at project level. The second one –the use case level Table- contained information about every use case (which is relevant, as a project is described as a set of use cases). It included almost the same variables–except for UFP-, but this time they were registered at use case level and two variables that characterize the project, which are CONTEXT AND WEB –which were previously described-, were added. Because of the size of this second table, which has 181 rows, it was not included in the article, but it can be found in [Robiolo 2009].

The relationship between these two tables prompted the use of the Multi-relational Data Mining model, since it represents a dependence "one to many" between the two tables. To run this model we allotted 70 % of the information to the training of the model -8 projects- and 30 % to its testing -3 projects-. This was so because 8 was the minimum number of projects that the tool requested to obtain significant models, which was considered to be a reasonable distribution of information. The tool made a *random selection* of the 3 projects to be tested: P1, P5 and P7.

The results obtained by the technique NN were compared to those reported by the authors in the above mentioned compilation, in order to verify if the error estimation had been improved.

3.1 NN Technique

In order to obtain the effort estimation of each of the three projects to be tested, we used the NN technique [Haykin1999]. First, we designed the 16 models described in Figure 2, so that we could analyze the incidence of each metric on the error estimation improvement. The variables used in each of these models are shown in Table 4. From each model, the three estimation values for P1, P5 and P7 were obtained.

M1, the starting model, was made up by all the variables at both levels: project and use case. Each one of the other 15 models was designed to test the way in which each variable helped to obtain the smallest estimation error. In M2 UFP was taken out. In M3 to M10, the estimation value of each metric (T, EO, P and AP) was calculated by using the two tables together (project level and use case level) and then by using only the project level table. This was done in order to determine the incidence of the use case levels in the error improvement. M11 and M12 used the set of three metrics -T, EO and AP- using the two tables together and the project level table alone. M13 did not include the values of the use case level table and M14 to M16 introduced variations to the model, with which the best result was obtained.

M1. Starting model.
M2. Incidence of UFP in the model, at project and use case levels.
M3. Use of T as a unit of measurement, at project and use case levels.
M4. Use of T as a unit of measurement, at project level.
M5. Use of P as a unit of measurement, at project and use case levels.
M6. Use of P as a unit of measurement, at project level.
M7. Use of EO as a unit of measurement, at project and use case levels.
M8. Use of EO as a unit of measurement, at project level.
M9. Use of AP as a unit of measurement, at project and use case levels.
M10. Use of AP as a unit of measurement, at project level.
M11. Use of T, EO, AP as units of measurement, at project and use case levels.
M12. Use of T, EO, AP as units of measurement, at project level.
M13. Incidence of the use case table as a whole
M14. Variations at project level in the best model (M11): without T
M15. Variations at project level in the best model (M11): without T, and adding UFP .
M16. Variations at project level in the best model (M11): without T and AP.

Fig. 2. Models defined for the technique NN

Table 4. Variables used in each model when employing the technique NN and MRE mean values obtained

Model	Project level					Use case level							Error Estimation
	EO	UFP	T	P	AP	EO	AP	T	P	Context	Web	AE	Mean MRE
M1	X	X	X	X	X	X	X	X	X	X	X	X	0.138
M2	X		X	X	X	X	X	X	X	X	X	X	0.131
M3			X					X		X	X	X	0.142
M4			X							X	X	X	0.142
M5				X					X	X	X	X	0.135
M6				X						X	X	X	0.135
M7	X					X				X	X	X	0.14
M8	X									X	X	X	0.125
M9					X		X			X	X	X	0.121
M10					X					X	X	X	0.123
M11	X		X		X	X	X	X		X	X	X	0.107
M12	X		X		X					X	X	X	0.114
M13	X	X	X	X	X								0.345
M14	X				X	X	X	X		X	X	X	0.12
M15	X	X				X	X	X		X	X	X	0.134
M16	X					X	X	X		X	X	X	0.129

Once the estimations had been made, they were compared to the AE of each project and the error obtained was expressed with MRE equations. The MRE mean are shown in Table 4.

3.2 NN Results Analysis

To better identify the best model which might help to obtain the most accurate effort estimation, all the variables in Table 3 and in the use case level Table were included in M1.

The variable UFP was taken away from M2 in order to verify if it was significant. The variation of the results obtained for M1 and M2 was minimal, with a slight improvement of the error in M2. Similar results were obtained in M14 when introducing UFP and in M11 when taking out AP. Results were improved in M16 by taking UFP away from M15. **Consequently, it can be stated that the use of UFP with the NN technique does not significantly improve the error in the estimation of effort**.

When analyzing the individual contributions offered by the units of measurement T, EO, P and AP at both project and use case levels (M3, M5, M7, M9), the best value was obtained by AP, however, none of them could improve the values obtained by M2. Besides, when extracting variables T, EO, P and AP at use case level in M4, M6, M8, M10, the variation registered was minimal if compared to M3, M5, M7, M9. For instance, the MRE mean in M3 was the same as M4. Thus, we infer that **what is important to improve the estimation is the information provided by the whole set of metrics rather than that provided by each metric in isolation.**

The best estimation result was obtained when using the NN technique in M11, in which a combination of the metrics T, EO, and AP, with the values registered at project and use case level, was employed. M12 was a modification of M11, in which the previously mentioned variables at use case level were removed. This did not improve the model; on the contrary, the negative difference was significant. Consequently, we may conclude that detail at use case level is important for the set of metrics T, EO and AP. Moreover, this result shows that P do not improve the estimation error, since P equal the sum of T plus AP.

To design M13, the use case level table was eliminated from M1, so the estimation was performed just by using the table at project level. The worst results were obtained with this model: the biggest error is shown in table 4. Thus, we may infer that **detail at use case level improves effort estimation**.

3.3 Comparison with reported results

To verify if it was possible to reduce the error in early effort estimation when employing the metrics T, EO and P –if compared to the reported errors above mentioned- the null and the alternative hypothesis were defined in the following way:

H0: For the set of MRE of the randomly selected projects, the median obtained when calculated with the NN technique will be equal to the median reported.

H1: For the set of MRE of the randomly selected projects, the median obtained when calculated with the NN technique will be smaller than the median reported.

As the number of models was no bigger than sixteen, the non-parametric sign test was selected to test these hypotheses because it does not require the assumption that the population is normally distributed. In many applications, when the normality assumption is questionable, this test is used instead of the one sample t-test. It is a less powerful alternative to the Wilcoxon signed ranks test, but it does not assume that the population probability distribution is symmetric [Montgomery and Runger 1996].

The median reported was calculated based on the data included in Robiolo (2009); its value was 0.23. H0 was rejected at an $\alpha = 0.005$ level of significance for the one sided test, as the number of positive values was 1, which is smaller or equal to the critical size value defined for the sample. The only case in which the MRE of Table 4 was bigger than 0.23 was for model M13, in which the use case label table was not included. This implies that for this case study, the effort estimation error was improved when the metrics T, EO and P were used with the NN technique, if the results are compared with the results reported in the survey.

Also, if the median reported is calculated with only the small projects, as the sample are also small projects, the median was 0.18. The H0 was rejected at the 0.01 level of significance.

To conclude, the answers to the research questions posed at the beginning of this study are:

a. The NN technique, when applied to a multi-relational model, in which the metrics T, EO and P were used, reduced the effort estimation error, when compared to the results reported in the survey.

b. It was extremely important to register the hours worked on each use case because the difference between the root mean square error in M11, in which all the variables at use case level where included, was 58% smaller than the error in model M13, in which no variable at use case level was included (cfr. M13 and M11 in Table 4).

4. Final Conclusion

The principal contribution of this work is that the accuracy of effort estimations was improved by using the variables T, EO and P at use case level detail, with a Multi-relational Data Mining technique.

It is also important to point out that we managed to test the use of a Data Mining technique (NN), in spite of the fact that the data was scarce (only 11 business projects). It was possible to overcome such limitation by employing detailed data (at use case level) of those projects. Thus, the requirements necessary to apply this type of technique (number of variables and number of instances) were fulfilled.

The fact that the NN technique could be supported by the SQL server tool, which is an accessible tool for the small and medium-sized software company, makes the results obtained more attractive. Besides, although the help of a Data Mining expert may be necessary to define the model, once it has been defined, no special knowledge is necessary to run it.

Evidently, the limited data used –just a set of 11 small business projects- does not allow us to generalize the conclusions. To do so, it would be necessary to build a new case study using projects of different types and sizes.

5. Acknowledgment

Our thanks to the Research Fund of Austral University, which made this study possible.

References

Albrecht, A.J. (1979). Measuring Application Development Productivity. Joint SHARE/GUIDE/IBM Application Development Symp.

Dzeroski, Saso. (2003). Multi-Relational Data Mining: An Introduction. ACM SIGKDD Exploration, Vol 5, issue 1, 1-16.

Fenton, N.E. and Pfleeger, S.L. (1997), Software Metrics, PWS Publishing Company.

Haykin, S. 1999. Neural Networks: A comprensive foundation. 2nd Edition, Prentice – Hall.

Kitchenham, B. (1997) "Counterpoint: the problem with Function Points", In: IEEE Software, Vol. 14, No. 2.

Lavazza L. and Robiolo, G. (2010). "Introducing the evaluation of complexity in functional size measurement: a UML-based approach", in Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM '10). ACM, New York, NY, USA, Article 25 , 9 pages.

McCabe, T. (1976). "A Complexity measure", *IEEE Transactions on software Engineering*, Vol. SE-2, NO. 4.

Montgomery D.C. and Runger G.C. (1996). Probabilidad y Estadísticas, aplicadas a la Ingeniería. Mc GRaw Hill.

Robiolo, G. (2008). "A Simple approach that improves early effort estimation based on use cases". Proceedings of the Third International Doctoral Symposium on Empirical Software Engineering IDoESE 2008, pages 19-24.

Robiolo, G., Orosco, R. (2008). "Employing use cases to early estimate effort with simpler metrics". *Innovations in Systems and Software Engineering*, Springer London, Vol.4 N 1, 31-43.

Robiolo, G. (2009), "Transacciones, Objetos de Entidad y Caminos: métricas de software, basadas en casos de uso, que mejoran la estimación temprana de esfuerzo". Tesis Doctoral UNLP. http://postgrado.info.unlp.edu.ar/Carrera/Doctorado/Tesis%20Doctorales.html

Gabriela Robiolo, Cristina Badano, and Ricardo Orosco (2009), "Transactions and paths: Two use case based metrics which improve the early effort estimation". In Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement (ESEM '09). IEEE Computer Society, Washington, DC, USA, 422-425.