Building an Expert-based Web Effort Estimation Model using Bayesian Networks

Emilia Mendes
Computer Science department
The University of Auckland
Private Bag, 92019, Auckland, NZ
emilia@cs.auckland.ac.nz

Carmel Pollino
The Fenner School of Environment and Society
The Australian National University
Canberra, Australia
carmel.pollino@anu.edu.au

Nile Mosley
MetriQ Ltd. (Research Division)
P.O. Box 837
Oneroa, Auckland, NZ
nile@metriq.biz

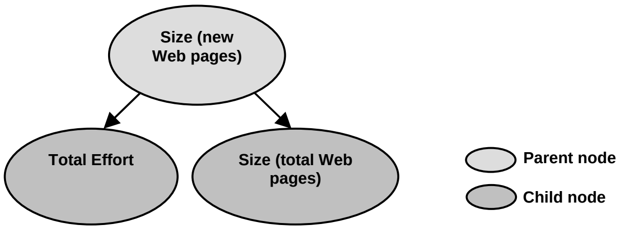
OBJECTIVE – The objective of this paper is to describe a case study where Bayesian Networks (BNs) were used to construct an expert-based Web effort model.
METHOD – We built a single-company BN model solely elicited from expert knowledge, where the domain expert was an experienced Web project manager from a small Web company in Auckland, New Zealand. This model was validated using data from eight past finished Web projects.
RESULTS – The BN model has to date been successfully used to estimate effort for four Web projects, providing effort estimates superior to those based solely on expert opinion.
CONCLUSIONS – Our results suggest that, at least for the Web Company that participated in this case study, the use of a model that allows the representation of uncertainty, inherent in effort estimation, can outperform expert-based estimates. Another five companies have also benefited from using Bayesian Networks, with very promising results.

Web effort estimation, Bayesian networks, single-company model, expert-based elicitation.

1. INTRODUCTION

A cornerstone of Web project management is effort estimation, the process by which effort is forecasted and used as basis to predict costs and allocate resources effectively, so enabling projects to be delivered on time and within budget. Effort estimation is a very complex domain where the relationship between factors is non-deterministic and has an inherently uncertain nature. E.g. assuming there is a relationship between development effort and an application's size (e.g. number of Web pages, functionality), it is not necessarily true that increased effort will lead to larger size. However, as effort increases so does the *probability* of larger size. Effort estimation is a complex domain where corresponding decisions and predictions require reasoning with uncertainty.
Within the context of Web effort estimation, numerous studies investigated the use of effort prediction techniques. However, to date only Mendes [15]-[18], and Mendes and Mosley [21] investigated the explicit inclusion, and use, of uncertainty, inherent to effort estimation, into models for Web effort estimation. Mendes [15]-[17] built a Hybrid Bayesian Network (BN) model (structure expert-driven and probabilities data-driven), which presented significantly superior predictions than the mean- and median-based effort [16], multivariate regression [15][17], case-based reasoning and classification and regression trees [17]. Mendes (2008) [18], and Mendes and Mosley [21] extended [15]-[17] by building respectively four and eight BN models (combinations of Hybrid and data-driven). These models were not optimised, as previously done in [15]-[17], which might have been the reason why they presented significantly worse accuracy than regression-based models.
A BN is a model that supports reasoning with uncertainty due to the way in which it incorporates existing complex domain knowledge [10]. Herein, knowledge is represented using two parts. The first, the qualitative part, represents the structure of a BN as depicted by a directed acyclic graph (digraph) (see Fig. 1). The digraph's nodes represent the relevant variables (factors) in the domain being modelled, which can be of different types (e.g. observable or latent, categorical). The digraph's arcs represent the causal relationships between variables, where

relationships are quantified probabilistically [10]. The second, the quantitative part, associates a node conditional probability table (CPT) to each node, its probability distribution. A parent node's CPT describes the relative probability of each state (value); a child node's CPT describes the relative probability of each state conditional on every combination of states of its parents (e.g. in Figure 1, the relative probability of Total effort (TE) being 'Low' conditional on Size (new Web pages) (SNWP) being 'Low' is 0.8). Each column in a CPT represents a conditional probability distribution and therefore its values sum up to 1 [10].



CPT for node Size (new Web pages)	
Low	0.2
Medium	0.3
High	0.5

CPT for node Total Effort (TE)			
Size (new Web pages)	***Low***	***Medium***	***High***
Low	0.8	0.2	0.1
Medium	0.1	0.6	0.2
High	0.1	0.2	0.7

FIGURE 1: A small BN model and two CPTs

Once a BN is specified, evidence (e.g. values) can be entered into any node, and probabilities for the remaining nodes automatically calculated using Bayes' rule [34]. Therefore BNs can be used for different types of reasoning, such as predictive and "what-if" analyses to investigate the impact that changes on some nodes have on others [8]. Within the context of Web effort estimation there are issues with building data-driven or hybrid Bayesian models, as follows:

1. Any dataset used to build a BN model should be large enough to provide sufficient data capturing all (or most) relevant combinations of states amongst variables such that probabilities can be learnt from data, rather than elicited manually. Under such circumstance, it is very unlikely that the dataset would contain project data volunteered by only a single company (single-company dataset). As far as we know, the largest dataset of Web projects available is the Tukutuku dataset (195 projects) [26]. This dataset has been used to build data-driven and hybrid BN models; however results have not been encouraging overall, and we believe one of the reasons is due to the small size of this dataset.
2. Even when a large dataset is available, the next issue relates to the set of variables part of the dataset. It is unlikely that the variables identified, represent all the factors within a given domain (e.g. Web effort estimation) that are important for companies that are to use the data-driven or hybrid model created using this dataset. This was the case with the Tukutuku dataset [21], even though the selection of which variables to use had been informed by two surveys [26]. However, one could argue that if the model being created is hybrid, then new variables (factors) can be added to, and existing variables can be removed from the model. The problem is that every new variable added to the model represents a set of probabilities that need to be elicited from scratch, which may be a hugely time consuming task.
3. Different structure and probability learning algorithms can lead to different prediction accuracy [21]; therefore one may need to use different models and compare their accuracy, which may also be a very time consuming task [21].
4. When using a hybrid model, the BN's structure should ideally be jointly elicited by more than one domain expert, preferably from more than one company, otherwise the model built may not be general enough to cater for a wide range of companies [21]. There are situations, however, where it is not feasible to have several experts from different companies cooperatively working on a single BN structure. One such situation is when the companies involved are all consulting companies potentially sharing the same market. This was the case within the context of this research.
5. Ideally the probabilities used by the data-driven or hybrid models should be revisited by at least one domain expert, once they have been automatically learnt using the learning algorithms available in BN tools. However, depending on the complexity of the BN model, this may represent having to check thousands of probabilities, which may not be feasible. One way to alleviate this problem is to add additional factors to the BN model in order to reduce the number of causal relationships reaching child nodes (divorcing [10]); however, all probabilities for the additional factors would still need to be elicited from domain experts.
6. The choice of variable discretisation, structure learning algorithms, parameter estimation algorithms, and the number of categories used in the discretisation all affect the accuracy of the results and there are no clear-cut guidelines on what would be the best choice to employ. It may simply be dependent on the dataset being used, the amount of data available, and trial and error to find the best solution [21].

Therefore, given the abovementioned constraints, as part of a NZ-government-funded project on using Bayesian Networks to Web effort estimation, we decided to develop several expert-based company-specific Web effort BN models, with the participation of numerous local Web companies in the Auckland region, New Zealand. The development and successful deployment of one of these models is the subject and contribution of this paper. Our contribution goes beyond the area of Web engineering given that the process presented herein can also be used to build BN models for non-Web companies. In addition, to our knowledge this is the first time that a study in either Web or Software Engineering describes the creation and use of an expert-based BN model; thus we believe this is also another contribution of this paper.

Note that we are not suggesting that data-driven and hybrid BN models should not be used. On the contrary, they have been successfully employed in numerous domains [36][42]; however the specific domain context of this paper – that of Web effort estimation, provides other challenges (described above) that lead to the development of solely expert-driven BN models.

We would also like to point out that in our view Web and software development differ in a number of areas, such as: Application Characteristics, Primary Technologies Used, Approach to Quality Delivered, Development Process Drivers, Availability of the Application, Customers (Stakeholders), Update Rate (Maintenance Cycles), People Involved in Development, Architecture and Network, Disciplines Involved, Legal, Social, and Ethical Issues, and Information Structuring and Design. A detailed discussion on this issue is provided in [25].

The remainder of the paper is organised as follows: Section 2 provides a brief overview of work in Web effort estimation, followed by the description of the overall process used to build and validate BNs in Section 3. Section 4 details the process, focusing on the expert-based Web Effort BN, and the validation of the model. Finally, conclusions and comments on future work are given in Section 5.

2. LITERATURE REVIEW OF WEB EFFORT ESTIMATION STUDIES

There have been numerous attempts to model effort estimation for Web projects. Table 1 presents a summary of previous studies. Whenever two or more studies compared different effort estimation techniques using the same dataset, we only included the study that used the greatest number of effort estimation techniques.

Mendes and Counsell [20] were the first to investigate this field by building a model that used machine-learning techniques with data from student-based Web projects, and size measures harvested late in the project's life cycle. Mendes and collaborators also carried out a series of consecutive studies [9],[19],[20]-[32] where models were built using multivariate regression and machine-learning techniques with data on industrial Web projects. Recently they also proposed and validated size measures harvested early in the project's life cycle, and therefore better suited to resource estimation [26].

Other researchers have also investigated resource estimation for Web projects. Reifer [38] proposed an extension of the COCOMO model, and a single size measure harvested late in the project's life cycle. None were validated empirically. This size measure was later used by Ruhe et al. [39], who further extended a software engineering hybrid estimation technique, named CoBRA© [3], to Web projects, using a small data set of industrial projects, mixing expert judgement and multivariate regression. Later, Baresi et al. [1][2], and Mangia et al. [14] investigated effort estimation models and size measures for Web projects based on a specific Web development method, namely the W2000. Costagliola et al. [4] compared two types of Web-based size measures for effort estimation. A detailed survey of Web cost estimation studies is given in [21].

As mentioned in the Introduction, five studies investigated the explicit inclusion, and use of uncertainty, inherent to effort estimation, into models for Web effort estimation. Mendes [15]-[17] used data on 150 projects from the Tukutuku dataset to built a Hybrid Bayesian Network (BN) model (structure expert-driven and probabilities data-driven). This model, which used a single training & validation set, presented significantly superior predictions than the mean- and median-based effort [16], multivariate regression [15],[17], case-based reasoning and classification and regression trees [17]. Later, Mendes [18], and Mendes and Mosley [21], using data on 195 project also from the Tukutuku dataset, extended [15]-[17] by building four (two Hybrid and two data-driven) and eight (four Hybrid and four data-driven) different BN models respectively. These models were benchmarked against stepwise regression, median and mean models, and none of the BN models presented significantly better accuracy than regression models. None of the BN models were optimised, contrary to what was previously done in [15]-[17]. We believe this might have contributed to their poor performance.

In summary, there is nearly an equal split between the use of data on student-based projects and industrial data; estimates obtained by applying Stepwise regression or Case-based reasoning techniques; and accuracy measured using MMRE, followed by MdMRE and Pred(25).

TABLE 1: Summary of Literature Review

Study	Type	# datasets - (# datapoints)	Projects in dataset(s)	Size Measures	Prediction techniques	Best technique(s)	Measure Prediction Accuracy
1st [20]	Case study	2 - (29 and 41)	2nd year Computer Science (CS) projects	Page Count, Reused Page Count, Connectivity, Compactness, Stratum, Structure	Case based reasoning, Linear regression, Stepwise regression	Case based reasoning for high experience group	MMRE
2nd [37]	Not detailed	1 - (46)	industrial projects	Web objects	WEBMO (parameters generated using linear regression)	-	Pred(n)
3rd [30]	Case study	1 - (37)	Honours (Hns) and postgraduate (PG) CS projects	Length size, Reusability, Complexity, Size	Linear regression Stepwise regression	Linear Regression	MMRE
4th [9]	Case study	1 - (37)	Hns and PG CS projects	Structure metrics, Complexity metrics, Reuse metrics, Size metrics	Generalised Linear Model	-	Goodness of fit

5th [29]	Case study	1 - (25)	Hns and PG CS projects	Requirements and Design measures, Application measures	Case-based reasoning		MMRE, MdMRE, Pred(25), Boxplots of absolute residuals
6th [32]	Case study	1 - (37)	Hns and PG CS projects	Page Count, Media Count, Program Count, Reused Media Count, Reused Program Count, Connectivity Density, Total Page Complexity	Case-based reasoning, Linear regression, Stepwise regression, Classification and Regression Trees	Linear/ stepwise regression or case-based reasoning (depends on the measure of accuracy employed)	MMRE, MdMRE, Pred(25), Boxplots of absolute residuals
7th [39]	Case study	1 - (12)	industrial projects	Web Objects	COBRA, Expert opinion, Linear regression	COBRA	MMRE, Pred(25), Boxplots of absolute residuals
8th [27]	Case study	2 - (37 and 25)	Hns and PG CS projects	Page Count, Media Count, Program Count, Reused Media Count (only one dataset), Reused Program Count (only one dataset), Connectivity Density, Total Page Complexity	Case-based reasoning	-	MMRE, Pred(25), Boxplots of absolute residuals
9th [2]	Formal experiment	1 - (30)	CS projects	Information, Navigation and Presentation model measures	Ordinary least squares regression	-	-
10th [12]	Not detailed	unknown	unknown	Functional, Navigational Structures, Publishing and Multimedia sizing measures	An exponential model named Metrics Model for Web Applications (MMWA)	-	-
11th [4]	Case study	1 – (15)	industrial projects	Web pages, New Web pages, Multimedia elements, New multimedia elements, Client side Scripts and Applications, Server side Scripts and Applications, All the elements that are part of the Web Objects size measure	Linear regression, Stepwise regression, Case-based reasoning, Classification and Regression Trees	All techniques provided similar prediction accuracy	MMRE, MdMRE, Pred(25), Boxplots of residuals, boxplots of z
12th [15], [16], [17]	Case study	1 – (150)	industrial projects	Total Web pages, New Web pages, Total Images, New Images, Features off-the-shelf (Fots), High & Low effort Fots-Adapted, High & Low effort New Features, Total High & Low Effort Features	Bayesian Networks, Stepwise Regression [15], [17], Mean and Median effort [16], Case-based reasoning [17], Classification and regression Trees [17]	Bayesian Networks provided superior predictions	MMRE, MdMRE, MEMRE, MdEMRE, Pred(25), Boxplots of residuals, boxplots of z
13th [18]	Case study	1 – (195)	industrial projects	Total Web pages, New Web pages, Total Images, New Images, Features off-the-shelf (Fots), High & Low effort Fots-Adapted, High & Low effort New Features, Total High & Low Effort Features	Bayesian Networks, Stepwise Regression, Mean and Median effort	Stepwise Regression provided superior predictions	MMRE, MdMRE, MEMRE, MdEMRE, Pred(25), Boxplots of residuals, boxplots of z
14th [17]	Case study	1 – (195)	industrial projects	Total Web pages, New Web pages, Total Images, New Images, Features off-the-shelf (Fots), High & Low effort Fots-Adapted, High & Low effort New Features, Total High & Low Effort Features	Bayesian Networks, Stepwise Regression, Mean and Median effort	Stepwise Regression provided superior predictions	MMRE, MdMRE, MEMRE, MdEMRE, Pred(25), Boxplots of residuals, boxplots of z

3. GENERAL PROCESS USED TO BUILD BNs

The BN presented in this paper was built and validated using an adaptation of the Knowledge Engineering of Bayesian Networks (KEBN) process proposed in [42] (see Figure 2). In Figure 2 arrows represent flows through the different processes, depicted by rectangles. Such processes are executed either by people – the Knowledge Engineer (KE) and the Domain Experts (DEs) (white rectangles), or by automatic algorithms (dark grey rectangles). Within the context of this paper the first author was the KE, and a Web project manager from a well-established Web company in Auckland was the DE.
The three main steps within the adapted KEBN process are the Structural Development, Parameter Estimation, and Model Validation. This process iterates over these steps until a complete BN is built and validated. Each of these three steps is detailed below:

Structural Development: This step represents the qualitative component of a BN, which results in a graphical structure comprised of, in our case, the factors (nodes, variables) and causal relationships identified as fundamental for effort estimation of Web projects. In addition to identifying variables, their types (e.g. query variable, evidence variable) and causal relationships, this step also comprises the identification of the states (values) that each variable should take, and if they are discrete or continuous. In practice, currently available BN tools require that continuous variables be discretised by converting them into multinomial variables [11], also the case with BN software used in this study. The BN's structure is refined through an iterative process. This structure construction process has been validated in previous studies [6][8][13][33][42] and uses the principles of problem solving employed in data modelling and software development [40]. As will be detailed later, existing literature in Web effort estimation, and knowledge from the domain expert were employed to elicit the Web effort BN's structure. Throughout this step the knowledge engineer(s) also evaluate(s) the structure of the BN, done in two stages. The first entails checking whether [11]: variables and their values have a clear meaning; all relevant variables have been included; variables are named conveniently; all states are appropriate (exhaustive and exclusive); a check for any states that can be combined. The second stage entails reviewing the BN's graph structure (causal structure) to ensure that any identified d-separation dependencies comply with the types of variables used and causality assumptions. D-separation dependencies are used to identify variables influenced by evidence coming from other variables in the BN [10][34]. Once the BN structure is assumed to be close to final knowledge engineers may still need to optimise this structure to reduce the number of probabilities that need to be elicited or learnt for the network. If optimisation is needed then techniques that change the causal structure (e.g. divorcing [10]) and the use of parametric probability distributions (e.g. noisy-OR gates [6][34]) are employed.

Parameter Estimation: This step represents the quantitative component of a BN, where conditional probabilities corresponding to the quantification of the relationships between variables [10][11] are obtained. Such probabilities can be attained via Expert Elicitation, automatically from data, from existing literature, or using a combination of these. When probabilities are elicited from scratch, or even if they only need to be revisited, this step can be very time consuming. In order to minimise the number of probabilities to be elicited some techniques to have been proposed in the literature [5][7][41]; however, as far as we are aware, there is no empirical evidence to date comparing their effectiveness for prediction, compared to probabilities elicited from scratch, using large and realistic BNs. This is one of the topics of our future work.

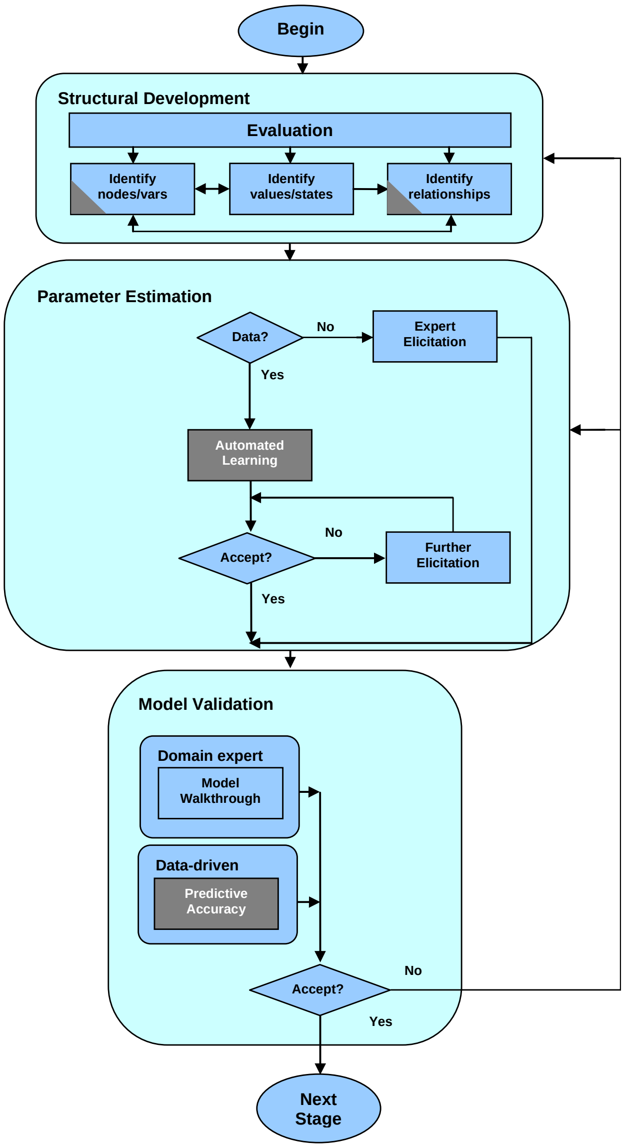


FIGURE 2: KEBN, adapted from [42]

Model Validation: This step validates the BN that results from the two previous steps, and determines whether it is necessary to re-visit any of those steps. Two different validation methods are generally used - Model Walkthrough and Predictive Accuracy.

Model walkthrough represents the use of real case scenarios that are prepared and used by domain experts to assess if the predictions provided by a BN correspond to the predictions experts would have chosen based on their own expertise. Success is measured as the frequency with which the BN's predicted value for a target variable (e.g. quality, effort) that has the highest probability corresponds to the experts' own assessment.

Predictive Accuracy uses past data (e.g. past project data), rather than scenarios, to obtain predictions. Data (evidence) is entered on the BN model, and success is measured as the frequency with which the BN's predicted value for a target variable (e.g. quality, effort) that has the highest probability corresponds to the actual past data. However, previous literature also documents a different measure of success, proposed by Pendharkar et al. [35], and later used by Mendes [15][18], and Mendes and Mosley [21]. Herein, an effort point forecast for each past project being used for validation is obtained by computing estimated effort as the sum of the probability (ρ) of a given effort scale point multiplied by its related mean effort (μ), after normalising the probabilities such that their sum equals one. Therefore, assuming that Estimated Effort is measured using a 5-point scale (Very Low to Very High), we have:

$$Estimated(Effort) = \rho_{VeryLow}\mu_{VeryLow} + \rho_{Low}\mu_{Low} + \rho_{Medium}\mu_{Medium} + \rho_{High}\mu_{High} + \rho_{VeryHigh}\mu_{VeryHigh} \quad (1)$$

Within the context of Web effort estimation and to some extent software effort estimation, the challenge using Predictive Accuracy is the lack of reliable effort data gathered by Web and Software companies. Most companies, who claim to collect effort data, use manually entered electronic timesheets (or even paper!) which is unreliable when staff rely on their memory and complete their timesheets at the end of the day. Collecting manually entered timesheets every 5 minutes (assume 1 minute/entry) in a bid to improve data accuracy increases data collection cost by as much as 10 fold. The problem here is that "effort accuracy" is inversely related to productivity, i.e., the longer one takes filling out timesheets the less time one has to do the real work!

4. PROCESS USED TO BUILD THE EXPERT-BASED WEB EFFORT BN

This Section revisits the adapted KEBN process (see Figure 2), detailing the tasks carried out for each of the three main steps that form part of that process. Before starting the elicitation of the Web effort BN model, the Domain Expert (DE) participating was presented with an overview of Bayesian Network models, and examples of "what-if" scenarios using a made-up BN. This, we believe, facilitated the entire process as the use of an example, and the brief explanation of each of the steps in the KEBN process, provide a concrete understanding of what to expect. We also made it clear that the knowledge Engineers were facilitators of the process, and that the Web company's commitment was paramount for the success of the process. The entire process took 18 hours to be completed, corresponding to 36 person hours in six 3-hour slots.

The DE who took part in this case study is the project manager (and owner) of a well-established Web company in Auckland (New Zealand). The company has one project manager, two developers employed by the company, and several sub-contractors. The project manager has worked in Web development for more than 10 years, and his company develops a wide range of Web applications, from static & multimedia-like to very large e-commerce solutions. They also use a wide range of Web technologies, thus enabling the development of Web 2.0 applications. Previous to using the BN model created, the effort estimates provided to clients would deviate from actual effort within the range of 10% to 40%.

<u>*Detailed Structural Development and Parameter Estimation*</u>: In order to identify the fundamental factors that the DE took into account when preparing a project quote we used the set of variables from the Tukutuku dataset [26] as a starting point (see Table 2). We first sketched them out on a white board, each one inside an oval shape, and then explained what each one meant within the context of the Tukutuku project. Our previous experience eliciting BNs in other domains (e.g. ecology) suggested that it was best to start with a few factors (even if they were not to be reused by the DE), rather than to use a "blank canvas" as a starting point.

TABLE 2: Tukutuku variables

	Variable Name	**Description**
Project Data	*TypeProj*	Type of project (new or enhancement).
	nLang	Number of different development languages used
	DocProc	If project followed defined and documented process.
	ProImpr	If project team involved in a process improvement programme.
	Metrics	If project team part of a software metrics programme.
	DevTeam	Size of a project's development team.
	TeamExp	Average team experience with the development language(s) employed.
Web application	*TotWP*	Total number of Web pages (new and reused).
	NewWP	Total number of new Web pages.
	TotImg	Total number of images (new and reused).
	NewImg	Total number of new images created.
	Num_Fots	Number of features reused without any adaptation.
	HFotsA	Number of reused high-effort features/functions adapted.
	Hnew	Number of new high-effort features/functions.
	TotHigh	Total number of high-effort features/functions
	Num_FotsA	Number of reused low-effort features adapted.
	New	Number of new low-effort features/functions.
	TotNHigh	Total number of low-effort features/functions

Within the context of the Tukutuku project, a new high-effort feature/function requires at least 15 hours to be developed by one experienced developer, and a high-effort adapted feature/function requires at least 4 hours to be adapted by one experienced developer. These values are based on collected data.

Once the Tukutuku variables had been sketched out and explained, the next step was to remove all variables that were not relevant for the DE, followed by adding to the white board any additional variables (factors) suggested by the DE. We also documented descriptions for each of the factors suggested.
Next, we identified the states that each factor would take. All states were discrete. Whenever a factor represented a measure of effort (e.g. Total effort), we also documented the effort range corresponding to each state, to avoid any future ambiguity. For example, 'very low' Total effort corresponded to 0+ to 8 person hours, etc.
Once all states were identified and documented, it was time to elicit the cause and effect relationships. As a starting point to this task we used a simple medical example from [10], and showed in Figure 3.

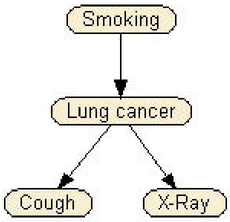


FIGURE 3: Medical example, adapted from [10]

This example clearly introduces one of the important points to consider when identifying cause and effect relationships – timeline of events. If smoking is to be a cause of lung cancer, it is important that the cause precedes the effect. This may sound obvious with regard to the example used; however, it is our view that the use of this simple example significantly helped the DE understand the notion of cause and effect, and how this related to Web effort estimation and the BN being elicited.
Once the cause and effect relationships were identified the Web effort BN's causal structure was as follows (see Figure 4). Note that Figure 4 is not a BN based directly on Table 2.

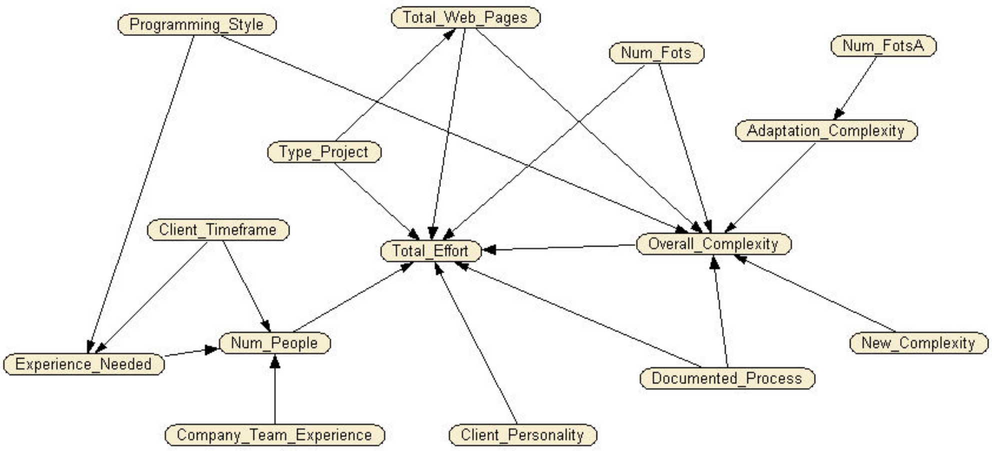


FIGURE 4: First BN causal model for the Web effort BN model

Nodes 'Total effort' and 'Overall Complexity' were each reached by a large number of relationships; therefore this structure needed to be simplified in order to reduce the number of probabilities to be elicited. New nodes were suggested by the KE (names ending in '_N', see Figure 5), and validated by the DE. The DE also made a few more changes to some of the relationships, leading to the BN causal structure presented in Figure 5.

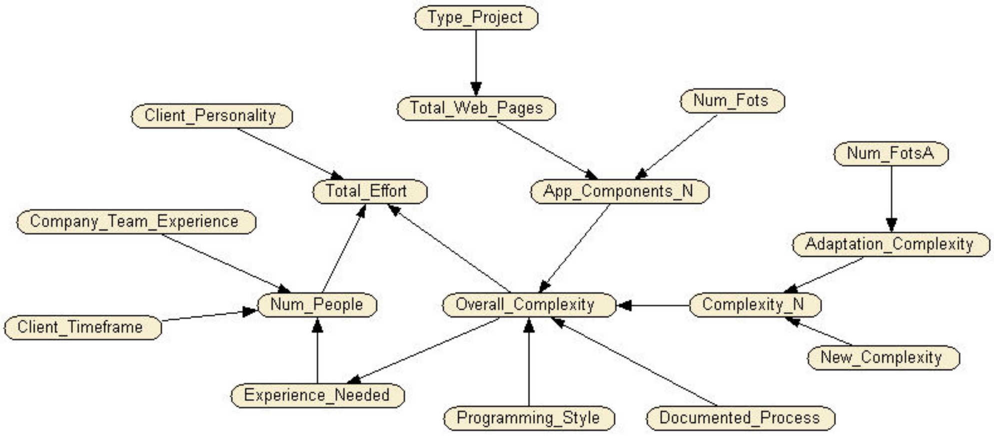


FIGURE 5: A new version of the BN causal model for the Web effort BN model

At this point the DE seemed happy with the BN's causal structure and the work on eliciting the probabilities was initiated. All probabilities were created from scratch, a very time consuming task (~8 to 10 hours).
While entering the probabilities, the DE decided to re-visit the BN's causal structure after revisiting his effort estimation process; therefore a new iteration of the Structural Development step took place. The final BN causal structure is shown in Figure 6. Here we present the BN using belief bars rather than labelled factors, so readers can see the probabilities that were elicited. Note that this BN corresponds to the current model being used by the Web company, which was also validated, to be detailed next.

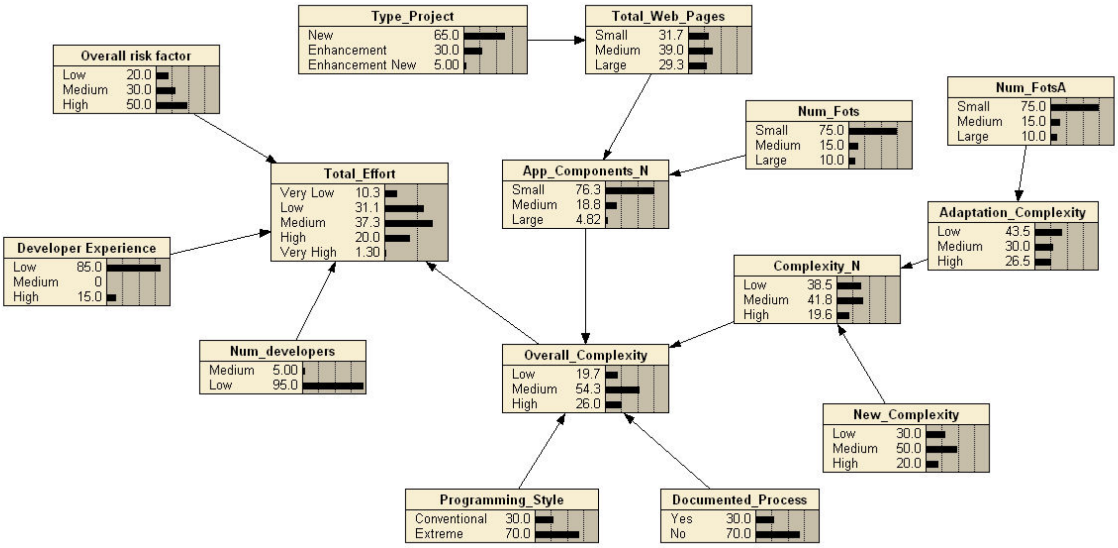


FIGURE 6: Final version of the BN causal model for the Web effort BN model

Detailed Model Validation: Both Model walkthrough and Predictive accuracy were used to validate the Web Effort BN model, where the former was the first type of validation to be employed. The DE used four different scenarios to check whether the node Total_effort would provide the highest probability to the effort state that corresponded to the DE's own suggestion. All scenarios were run successfully; however it was also necessary to use data from past projects, for which total effort was known, in order to check the model's calibration. A validation set containing data on eight projects was used. The DE selected a range of projects presenting different sizes and levels of complexity, where all eight projects were representative of the types of projects developed by the Web company: four were small projects; three were medium and one was large.

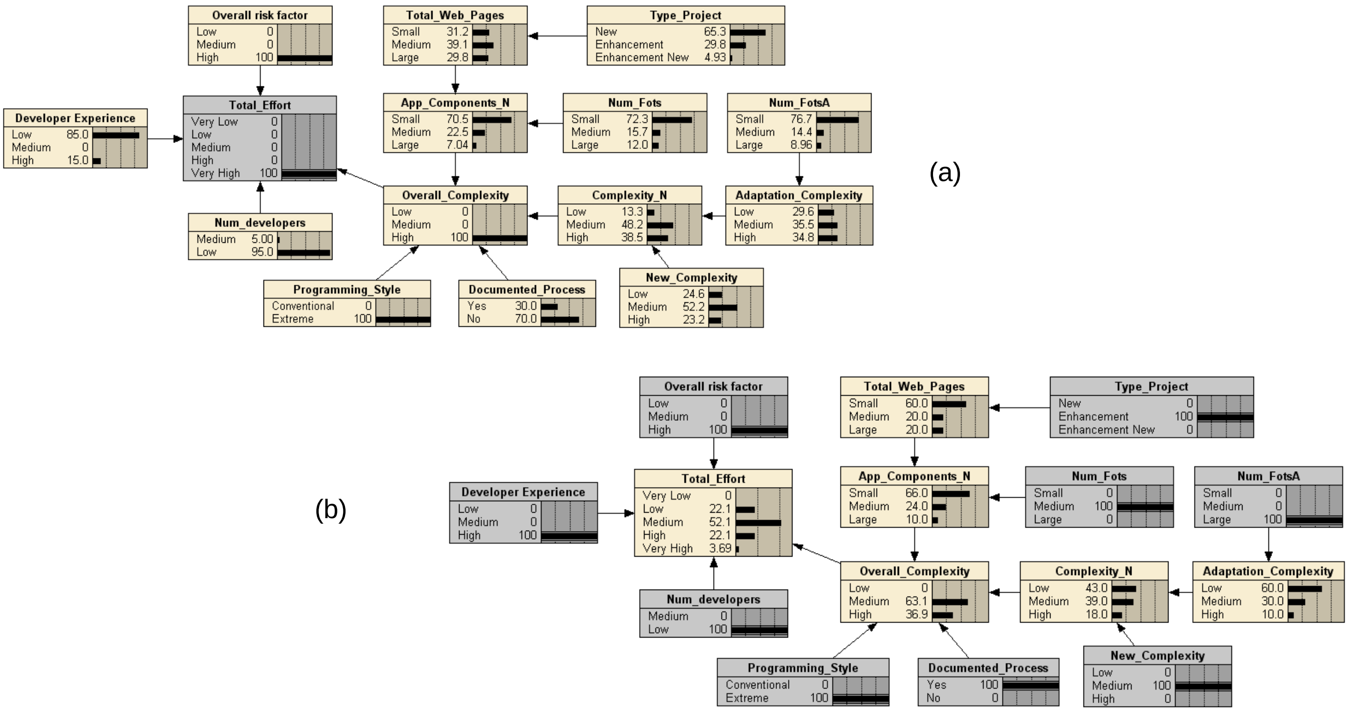


FIGURE 7: Diagnostic and Predictive scenarios using the Web effort BN model

For each project, evidence was entered in the BN model (in a similar way to Figure 7.b), and the effort range corresponding to the highest probability provided for 'Total Effort' was compared to that project's actual effort. For example, in Figure 7.b, this would correspond to 'Total Effort' = Medium. The company had also defined the range of effort values associated with each of the categories used to measure 'Total Effort'. In the case of the company described herein, Medium effort corresponds to 25 to 50 person hours.

Whenever actual effort did not fall within the effort range associated with the category with the highest probability, there was a mismatch; this meant that some probabilities needed to be adjusted. In order to know which nodes to target first we used a Sensitivity Analysis report, which provided the effect of each parent node upon a given query node. Within our context, the query node was 'Total Effort'.

Whenever probabilities were adjusted, we re-entered the evidence for each of the projects in the validation set that had already been used in the validation step to ensure that the calibration already carried out had not affected. This was done to ensure that each calibration would always be an improved upon the previous one. Once all eight projects were used to calibrate the model the DE assumed that the Validation step was complete.

Figure 7 shows two scenarios of use for the Web effort estimation BN. The first (Figure 7(a) shows the likely probabilities for all factors in the BN given an expected Total_effort = Very High (grey node in Figure 7); conversely, the second scenario shows the likely probabilities for Total_effort when evidence is entered along the BN (grey nodes).

This BN model has been in production for four months and has been successfully used to estimate effort on four projects, ranging from medium to large. This means that only the model has been used to obtain effort estimates, rather than relying on the expert's knowledge of previous projects.

We believe that the successful development of the Web effort BN model was greatly influenced by the commitment of the company, and also by the DE's very good knowledge and experience on estimating effort.

5. CONCLUSIONS

This paper has presented a case study where a Bayesian Model for Web effort estimation was built using solely knowledge of a Domain Expert from a well-established Web company in Auckland, New Zealand. This model was developed using the knowledge engineering for Bayesian Networks process (see Figure 2). Its causal structure went through two versions, because as the work progressed the expert's views on which factors were fundamental when he estimated effort also matured. Each session with the domain expert lasted for no longer than 3 hours. The final BN model was calibrated using data on eight past projects. These projects represented typical projects developed by the company, and believed by the expert to provide enough data for model calibration.

Since the model's adoption, it has been successfully used to provide effort quotes for four Web projects. Here effort estimates were obtained using solely the BN model.

The entire process used to build and validate the BN model took ~18 hours, where the largest amount of time was spent eliciting the probabilities. This is an issue to those building BN models from domain expertise only, and is currently the focus of our future work.

The elicitation process enables experts to think deeply about their effort estimation process and the factors taken into account during that process, which in itself is already advantageous to a company. This has been pointed out to us not only by the domain expert whose model is presented herein, but also by other companies with which we are working on model elicitations.

To date we have completed the elicitation of seven expert-driven Bayesian Models for Web effort estimation and have merged their causal structures in order to identify common Web effort predictors, and causal relationships. The reporting of this work will be the subject of a future publication.

6. ACKNOWLEDGMENTS

We would like to thank the DE for participating in this case study, and for his commitment to the research. This work is sponsored by the Royal Society of New Zealand, under the Marsden Fund research grant 06-UOA-201 MIS.

REFERENCES

[1] Baresi, L., S. Morasca, and P. Paolini (2002) An empirical study on the design effort for Web applications, *Proceedings of WISE 2002*, pp. 345-354.

[2] Baresi, L., S. Morasca, and P. Paolini (2003) Estimating the design effort for Web applications, *Proceedings of Metrics 2003*, pp. 62-72.

[3] Briand, L.C., K. El Emam, F. Bomarius (1998) COBRA: A Hybrid Method for Software Cost Estimation, Benchmarking and Risk Assessment, *Proceedings of the 20th ICSE*, 1998, pp. 390-399.

[4] Costagliola, G., Martino, S. Di, Ferrucci, F., Gravino, C., Tortora, G., and Vitiello, G. (2006) Effort estimation modeling techniques: a case study for web applications, *Procs. Intl. Conference on Web Engineering* (ICWE'06), pp. 9-16.

[5] Das, B. (2004) Generating Conditional Probabilities for Bayesian Networks: Easing the Knowledge Acquisition Problem, *arxiv.org/pdf/cs/0411034v1* (accessed in 2008).

[6] Druzdzel, M.J., and L.C. van der Gaag (2000) Building Probabilistic Networks: Where Do the Numbers Come From?, *IEEE Trans. on Knowledge and Data Engineering*, 12(4), 481-486.

[7] Fenton, N., M. Neil, and J. Caballero (2007), Using Ranked Nodes to Model Qualitative Judgments in Bayesian Networks, *IEEE Transactions on Knowledge and Data Engineering*, 19(10), pp. 1420-1432.

[8] Fenton, N., W. Marsh, M. Neil, P. Cates, S. Forey, and M. Tailor (2004) Making Resource Decisions for Software Projects, *Proc. ICSE'04*, pp. 397-406.

[9] Fewster, R., and E. Mendes (2001) Measurement, Prediction and Risk Analysis for Web Applications, *Proceedings of IEEE Metrics Symposium*, pp. 338 – 348.

[10] Jensen, F. V. (1996) *An introduction to Bayesian networks*. UCL Press, London.
[11] Korb, K.B., and A.E. Nicholson (2004) *Bayesian Artificial Intelligence*, CRC Press, USA.
[12] Mangia, L., and R. Paiano (2003) MMWA: A Software Sizing Model for Web Applications, Proc. Fourth International Conference on Web Information Systems Engineering, pp. 53-63.
[13] Mahoney, S.M., and K.B. Laskey (1996). Network Engineering for Complex Belief Networks, *Proc. Twelfth Annual Conference on Uncertainty in Artificial Intelligence*, pp. 389-396.
[14] Mangia, L., and R. Paiano (2003). MMWA: A Software Sizing Model for Web Applications, *Proc. Fourth International Conference on Web Information Systems Engineering*, pp. 53-63.
[15] Mendes, E., (2007a). E. Mendes, Predicting Web Development Effort Using a Bayesian Network, Proceedings of EASE'07, pp. 83-93.
[16] Mendes, E. (2007b), The Use of a Bayesian Network for Web Effort Estimation, Proceedings of International Conference on Web Engineering, pp. 90-104, LNCS 4607.
[17] Mendes, E. (2007c), A Comparison of Techniques for Web Effort Estimation, Proceedings of the ACM/IEEE International Symposium on Empirical Software Engineering, pp. 334-343.
[18] Mendes, E. (2008), The Use of Bayesian Networks for Web Effort Estimation: Further Investigation, Proceedings of ICWE’08, pp. 2-3-216.
[19] Mendes, E., and B.A. Kitchenham (2004). Further Comparison of Cross-company and Within-company Effort Estimation Models for Web Applications, *Proc. IEEE Metrics*, pp. 348-357.
[20] Mendes, E., and S. Counsell (2000). Web Development Effort Estimation using Analogy, *Proc. 2000 Australian Software Engineering Conference*, pp. 203-212.
[21] Mendes, E., and N. Mosley (2008), Bayesian Network Models for Web Effort Prediction: a Comparative Study, Transactions on Software Engineering, Vol. 34, Issue: 6, Nov/Dec 2008, pp. 723-737.
[22] Mendes, E., and N. Mosley (2005) Web Cost Estimation: principles and applications. *Web Engineering – Principles and Techniques*, Idea Group, Inc., Mehdi Khosrow-Pour and Jan Travers (Eds.), pp. 182-202.
[23] Mendes, E., and N. Mosley (2002). Further Investigation into the Use of CBR and Stepwise Regression to Predict Development Effort for Web Hypermedia Applications, *Proc. ACM/IEEE ISESE*, Nara, Japan, pp. 79-90.
[24] Mendes, E., Mosley, N. and Counsell, S. (2005a) Web Effort Estimation, *Web Engineering*, Springer-Verlag, Mendes, E. and Mosley, N. (Eds.) ISBN: 3-540-28196-7, pp. 29-73.
[25] Mendes, E., Mosley, N. and Counsell, S. (2005b) The Need for Web Engineering: An Introduction, *Web Engineering*, Springer-Verlag, Mendes, E. and Mosley, N. (Eds.) ISBN: 3-540-28196-7, pp. 1-28.
[26] Mendes, E., N. Mosley, and S. Counsell (2005c). Investigating Web Size Metrics for Early Web Cost Estimation, *Journal of Systems and Software,* 77(2), 157-172.
[27] Mendes, E., N. Mosley, and S. Counsell (2003). A Replicated Assessment of the Use of Adaptation Rules to Improve Web Cost Estimation, *Proc. ISESE*, pp. 100-109.
[28] Mendes, E., N. Mosley, and S. Counsell (2003). Early Web Size Measures and Effort Prediction for Web Costimation, *Proceedings of the IEEE Metrics Symposium*, pp. 18-29.
[29] Mendes, E., N. Mosley, and S. Counsell (2002). Comparison of Length, complexity and functionality as size measures for predicting Web design and authoring effort, *IEE Proc. Software*, 149(3), June, 86-92.
[30] Mendes, E., N. Mosley, and S. Counsell (2001). Web metrics - Metrics for estimating effort to design and author Web applications. *IEEE MultiMedia*, January-March, 50-57.
[31] Mendes, E., N. Mosley, and I. Watson (2002). A Comparison of Case-Based reasoning Approaches to Web Hypermedia Project Cost Estimation, *Proc. WWW’02*.
[32] Mendes, E., I. Watson, C. Triggs, N. Mosley, and S. Counsell (2003). A Comparative Study of Cost Estimation Models for Web Hypermedia Applications, *ESE*, 8(2), 163-196.
[33] Neil, M., Fenton, N., and Nielsen, L. (2000) Building Large-scale bayesian networks, *The knowledge Engineering Review*, KER, Vol. 15, No. 3, 257-284.
[34] Pearl J. (1988). *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann, San Mateo, CA.
[35] Pendharkar, P.C., Subramanian, G.H., and Rodger, J.A. (2005) A Probabilistic Model for Predicting Software Development Effort, *IEEE Trans. Software Eng*. Vol. 31, No. 7, 615-624.
[36] Pollino, C., A. White, and B.T. Hart (2007). Development and application of a Bayesian decision support tool to assist in the management of an endangered species. *Ecological Modelling* 201, 37-59.
[37] Reifer, D.J. (2000) Web Development: Estimating Quick-to-Market Software, *IEEE Software*, Nov.-Dec., 57-64.
[38] Reifer, D.J. (2002). Ten deadly risks in Internet and intranet software development, *IEEE Software*, Mar-Apr, 12-14.
[39] Ruhe, M., R. Jeffery, and I. Wieczorek (2003). Cost estimation for Web applications, *Proceedings ICSE 2003*, 285-294.
[40] Studer, R., V.R. Benjamins, and D. Fensel (1998). Knowledge engineering: principles and methods. *Data & Knowledge Engineering*, vol. 25, 161-197.
[41] Tang, Z., and B. McCabe (2007). Developing Complete Conditional Probability Tables from Fractional Data for Bayesian Belief Networks, Journal of Computing in Civil Engineering, 21(4), pp. 265-276.
[42] Woodberry, O., A. Nicholson, K. Korb, and C. Pollino (2004). Parameterising Bayesian Networks, *Proc. Australian Conference on Artificial Intelligence*, pp. 1101-1107.