GENERATING CONSTRUCTION KNOWLEDGE WITH KNOWLEDGE DISCOVERY IN DATABASES

Lucio Soibelman[1] and Hyunjoo Kim[2]

ABSTRACT

As the construction industry is adapting to new computer technologies in terms of hardware and software, computerized construction data becomes increasingly available. Knowledge Discovery in Databases (KDD) and Data Mining (DM) are tools that allow us to identify novel patterns in construction projects through analyzing the large amount of construction project data. Those technologies combine techniques from machine learning, artificial intelligence, pattern recognition, statistics, databases and visualization to automatically extract concepts, interrelationships, and patterns of interest from large databases. This paper presents both the steps required for the implementation of KDD and DM tools on large construction database and one case study demonstrating the feasibility of the proposed approach. In order to test the feasibility of the proposed approach, a prototype of Knowledge Discovery in Databases (KDD) system was developed and tested with a database, RMS (Resident Management System), provided by the US Corps of Engineers.

KEY WORDS

Knowledge Discovery in Databases (KDD), Data Mining, Machine Learning, Lean Construction, Knowledge, Decision Trees, Neural Networks

[1] Assistant Professor, Dept. of Civil Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801

[2] Ph.D. Candidate, Dept. of Civil Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801

INTRODUCTION

Past experience often plays a very important role in construction management. “How much time is this task going to take?” or “How many nails do we need to build this panel?” are the types of questions that project managers face daily in their planning activities. Failure or success in developing good schedules, budgets and other project management tasks depends on the project manager's ability to obtain reliable information in order to be able to answer these types of questions. Students and young practitioners tend to rely on information that is a regional average provided by various publishing companies. This is in contrast to experienced project managers who tend to rely heavily on their personal experience. Another aspect of construction management is that many researchers study one narrow topic in great detail, seeking to improve the available scheduling algorithms, estimating spreadsheets and other project management tools. Such a “micro-scale” level of research is important in providing the required tools for the project manager's tasks. However, even with the best scheduling tools, for example, low quality input information will in most cases produce inaccurate construction schedules as output. Thus, it is also important to have a broad approach of research in a "macro-scale" level.

These days, the construction industry is seeing explosive growth in its capabilities to both generate and collect data. Advances in scientific data collection, the introduction of bar codes for almost all commercial products, and computerization have generated a flood of data. Advances in data storage technology, such as faster, higher capacity, and cheaper storage devices (e.g. magnetic disks, CD-ROMS), better database management systems, and data warehousing technology, have allowed the transformation of this enormous amount of data into computerized database systems. As the construction industry is adapting to new computer technologies in terms of hardware and software, computerized construction data are becoming more and more available. However, in most cases, these data may not be used, or even properly stored. Several reasons exist: (i) construction managers do not have sufficient time to analyze the computerized data, (ii) complexity of the data analysis process is sometimes beyond the simple application, and (iii) up to now, there is no well defined automated mechanism to extract, preprocess and analyze the data and summarize the results so that the site managers can use it. On the other hand, it is obvious that valuable information can be obtained from an appropriate use of this data.

There is a need to analyze this increasing amount of available data and Knowledge Discovery in Databases (KDD) can be applied as a powerful tool to identify causal relationships in construction projects. Due to its own nature of variability, construction data differ even in a similar project. Some lean construction researchers suggested reducing construction variability by identifying and eliminating causes for possible deviations. Mining data will enable us to understand how systems that were once thought to be completely chaotic actually have predictable patterns (Peitgen 1992). Thus, although chaotic data and systems appear to be random in the construction project, there are patterns beneath the random behavior. Through KDD, pattern behind apparent random nature of construction projects can be determined. By applying Knowledge Discovery in Databases (KDD) to identify novel patterns, project managers will be able to build a knowledge models that may be used for the recurrent activities of on-going construction projects as well as for a future project activities.

To test the feasibility of the proposed approach a prototype of KDD system was developed and tested with RMS data provided by the US Corps of Engineers. RMS is an

automated construction management/ quality information system that is PC-based, LAN-compatible and oriented to the daily requirements of USACE (US Army Corps of Engineers).

RELEVANT ISSUES

AVAILABILITY AND RELEVANCE OF VERY LARGE DATABASES

Data management started about three decades ago, when no data specific information was explicitly stored along with the data. Often data had to be stored more than once across the organization leading to inconsistencies and inefficiencies. Data Management Systems were introduced in the late 1960's largely triggered by the Space Race. Constraints, such as data types, value ranges, dependencies, or generation languages were provided to ease application development at this time.

Nowadays the explosive growth of many business, government, and scientific databases has far outpaced our ability to interpret and digest the data. Such volumes of data clearly overwhelm the traditional methods of data analysis such as spreadsheets and ad-hoc queries. The traditional methods can create informative reports from data, but cannot analyze the contents of those reports. A significant need exists for a new generation of techniques and tools with the ability to automatically assist humans in analyzing the mountains of data for useful knowledge.

Historically the notion of finding useful patterns in raw data has been given various names, including knowledge extraction, information discovery, information harvesting, data archeology, and data pattern processing. By the end of 1980s, a new term, knowledge discovery in databases (KDD), was coined to replace all of the old terms whose objective was to find patterns and similarities in raw data. Artificial intelligence and machine learning practitioners quickly adopted KDD and used it to cover the overall process of extracting knowledge from databases. The term, Data Mining has been used in this context for the process when the mining algorithms were applied. Recently, as a result of the increasing attention of vendors and the popular trade press in this area, the words data mining have been adapted and have come to mean, like KDD, the overall process of extracting knowledge from databases.

CAUSAL ANALYSIS

Traditional approaches to causal analysis

The quality problems in the construction industry are very costly. Several papers recognized the significance of quality problems surrounding construction industry and tried to identify the causes of the quality problems (Arditi et al. 1998; Burati et al. 1992; Davis et al. 1989) where those researches concluded that major factors to causing the construction quality problems would be inadequate information, poor communication, poor care in workmanship, and lack of site supervision. However, the conclusions above may be very subjective and therefore, become a matter of judgment.

This paper proposes that Knowledge Discovery in Databases (KDD) be used to identify the causal relationships in a construction project by analyzing the large amount of data and also analyze the interaction between construction activities. Identifying the causal relationships in construction tasks through KDD will help the construction manager to take the appropriate courses of action required for recurrent activities of ongoing construction projects as well as for future projects.

Causal analysis model

Figure 1. General Systems Model

The general causal analysis model concentrates on three types of variables: inputs, processes, and outputs as shown in Figure 1. Inputs may include factors that can be internal factors (labors, resources, cost, legal or regulatory requirements and so on) as well as external factors (economics, weather and so on). The processes include activities: formal and planned functions, and informal and unplanned functions. The outputs may be intended and unintended, positive and negative and short-term and long-term.

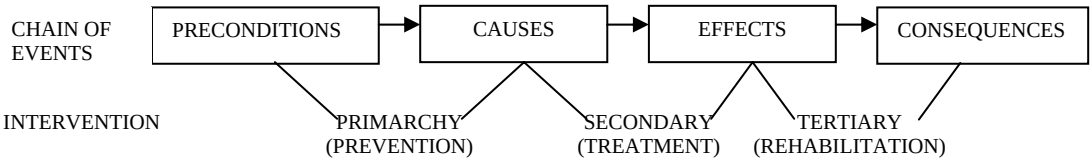

Figure 2. Suchman's Causal Models (Suchman, 1967)

Suchman's model (Figure 2) can be used to display the relationships among input, processes and outputs in causal analysis model where intervening in the chain of events is represented in three different ways: primary intervention (enables prevention at the root); secondary intervention (enables to reduce the effects); tertiary intervention (enables rehabilitation or reduction in the consequences). If the relationship between the precondition and causal variables is understood, many undesirable outcomes can be prevented even before they occur.

For example, referring to Figure 2, interventions of concrete problems can be described in three ways using the chain of events as follows:

- Primary Prevention: avoiding pouring of concrete on freezing weather,
- Secondary Treatment: using the right water/cement ratio, controlling slump, providing the required number of joints, and curing properly,
- Tertiary Rehabilitation: reinforcing or re-building the concrete structure in which the quality is lower than expected.

CASE STUDY - IMPLEMENTING KDD FOR DISCOVERING PATTERNS

In this section, a case study of a KDD on large construction database is presented with a sequence of five processes shown in Figure 3, the goal of which is to present the steps required during the KDD process and the type of knowledge that could be generated with the available tools. The initial data survey for a project in Fort Wayne, Indiana, provided by US Corps of Engineers demonstrated that one activity called "Installation of drainage pipelines" was behind schedule in 54% instances.

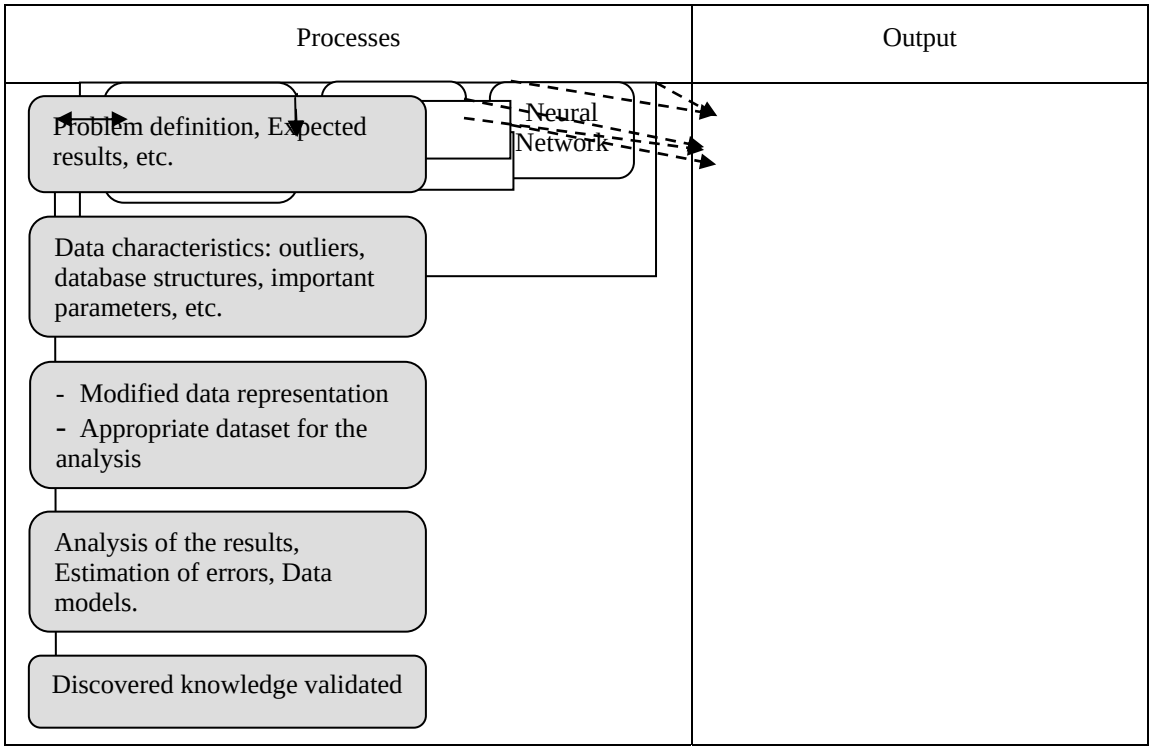

Figure 3. The main processes of the discovery approach

DATA PREPARATION

To date, most modern KDD tools have focused almost exclusively on building models. However, data preparation is a very important process since data itself may have been collected in an ad hoc manner, unfilled fields in records may be found, or mistakes in data entry may have been made. As a result, KDD process cannot succeed without a serious effort to prepare the data. Without the data discovery phase, the analyst will have no idea if the data quality can support the task at all. Once the quality and details are assessed, serious work is usually needed to get the data in shape for analysis.

Like any other real world applications, RMS data also has several problems such as missing parameter values, improper data types, out-of-range data, incomplete records or instances, and unavailable data. One benefit of data preparation is that it prepares both the data and the analyst. When data is properly prepared, the analyst gains understanding and insight into the content, range of applicability, and limits of the data. When data is correctly prepared and surveyed, the quality of the models produced will depend mostly on the content of the data, not so much on the ability of the analyst.

DATA MINING

Given the application characteristics presented in the previous section, the goal of this section is to develop an overall data analysis methodology that can be applied to find patterns that explain or predict any behaviors in construction projects. Three processes compose the approach in this section. In this case study, Feature Subset Selection was first used to calculate the relevance of features. Then, Decision Tree was used to extract rules from the data sets. Rules from Decision Tree made the input selection for the Neural Network a simple task and the understanding of outputs of Neural Network easier. In the

whole process, statistics played an important role in data validation and prediction. Finally, data/knowledge validation was sought to give users a good understanding of the patterns extracted from the database.

- Feature Subset Selection

The technique of Feature Subset Selection was required in the case study because many different attributes were available in the data set and was not obvious which features could be useful for the current problem. Also, practical machine learning algorithms, including decision tree algorithms such as ID3, C4.5 and CART are known to degrade in performance when faced with many features that are not necessary for predicting the desired output. The feature subset selection algorithm conducts a search for a good subset using the induction algorithm as part of the evaluation function. The accuracy of the induced classifiers is estimated using accuracy estimation techniques. The wrapper approach (Kohavi 1994) is well known in the machine learning community because of its accurate evaluation and was used in this application. There are two well-known induction algorithms such as the Decision Tree and the Naïve-Bayes induction algorithms. In Decision Tree, the tree is constructed by finding the best single-feature test to conduct at the root node of the tree. The Naïve-Bayesian classifier uses Bayes' rules to compute the probability of each class given the instance, assuming the features are conditionally independent.

- Decision Trees

Decision Tree is a tree-based knowledge representation methodology used to represent classification rules. The leaf nodes represent the class labels while non-leaf nodes represent the attributes associated with the objects being classified. The branches of the tree represent each possible value of the decision node from which they originate. Decision Trees are useful, particularly for solving problems that can be cast in terms of producing a single answer in the form of a class name. Based on answers to the questions at the decision nodes, one can find the appropriate leaf and the answer it contains. C4.5 (Quinlan 1993) is an example that uses the algorithms above. The first stage of C4.5 generates a decision tree. Each level of the decision tree represents a split of the data set. This split is chosen by examining each possible split of the data on each attribute, and choosing the one which best splits the data (according to an information theoretic measure of the distribution of classes in each subset). This continues for each level of the decision tree until there is no benefit from further segmenting the data. Once this has been done, rules are generated by traversing each branch of the tree and collecting the conditions at each branch of the decision tree. Each generated rule has a confidence percentage associated with the class it predicts. The uncertainty is caused by the generalization process, as some leaves on a tree may no longer contain single labels.

- Neural Networks (NN)

The foundation of the neural networks paradigm was laid in the 1950s and NN has gained significant attentions in the past decade due to the development of more powerful hardware and neural algorithms (Rumelhart 1994). Neural networks have been adopted in various engineering, business, military, and biomedical domains.

Among the numerous artificial neural networks which have been proposed, Backpropagation Networks have been extremely popular for their unique learning

capability. Backpropagation Networks (Rumelhart et al 1986) are layered, feed-forward models. Activations flow from the input layer through the hidden layer, then to the output layer. A Backpropagation Network typically starts out with a random set of weights. The network adjusts its weights each time it sees an input-output pair. Each pair is processed at two stages, a forward pass and a backward pass. The forward pass involves presenting a sample input to the network and letting activations flow until they reach the output layer. During the backward pass, the network's actual output is compared with the target output and error estimates are computed for the output units. The weights connected to the output units are adjusted in order to reduce the errors (a gradient descent method). The error estimates of the output units are then used to derive error estimates for the units in the hidden layer. Finally, errors are propagated back to the connections stemming from the input units. The Backpropagation Network updates its weights incrementally until the network stabilizes.

DATA ANALYSIS

- **Results from AI Decision Tree**

Figure 4 shows each level of the Decision Tree built with data from the project in Fort Wayne, Indiana. Interesting patterns can be found as follows. Each box in the tree in Figure 4 represents a node. The top node is called the root node. A decision tree grows from the root node, so the tree can be thought as growing upside down, splitting the data at each level to form new nodes. The resulting tree comprises many nodes connected by branches. Nodes that are at the end of branches are called leaf nodes and play a special role when the tree is used for prediction. In Figure 4, each node contains information about the number of instances and percentages at that node, and about the distribution of dependent variable values. The instances at the root node are all of the instances in the training set. This node contains 224 instances, of which 54 percent are instances of delay and 46 percent are of no delay. Below the root node (parent) is the first split that, in this case, splits the data into two new nodes (children) based on whether Inaccurate Site Survey is yes or no. The rightmost node (Yes of Inaccurate Site Survey) resulting from this split contains 36 instances, all of which are associated with schedule delays. Because all instances have the same value of the dependent variable, this node is considered pure and will not be split further. The leftmost node in the first split contains 188 instances, 46 percent of which are schedule delays. The leftmost node is then further split based on the value of Shortage of Equipment. An induction algorithm determines the order of the splits, Inaccurate Site Survey, Shortage of Equipment, and so on.

A tree that has only pure leaf nodes is called a pure tree, a condition that is not only unnecessary but is usually undesirable. Most trees are impure, that is, their leaf nodes contain cases with more than one outcome. Figure 4 reveals the following interesting patterns:

- Weather considered responsible for delays by site managers, appear not to be the most important cause in determining delays.
- Activities with Inaccurate Site Surveys are always delayed in the schedule.
- Also, Shortage of Equipment, Seasons, and Incomplete Drawing are very significant factors in determining schedule delay since induction algorithm tried to prioritize its splits by choosing the most significant split first.
- Once the Decision Tree is built, the tree can be used for predicting a new case by starting at the root (top) of the tree and following a path down the branches until a

leaf node is encountered. The path is determined by imposing the split rules on the values of the independent variables in the new instance. Navigating a tree to produce predicted values can become cumbersome as trees increase in size and complexity. It is possible to derive a set of rules for a tree with one rule for each leaf node simply by following the path between the root and that leaf node.

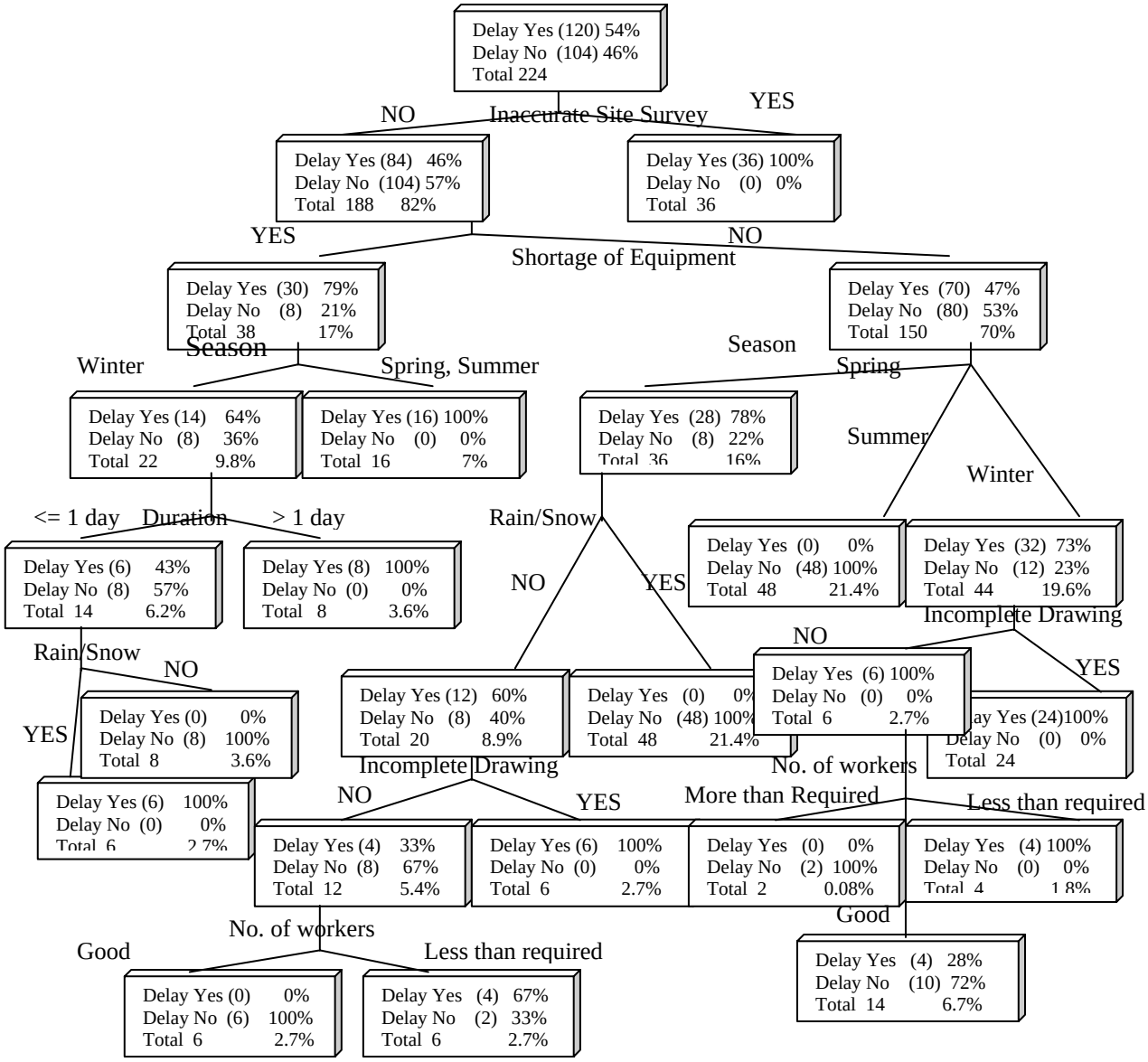

Figure 4. Decision Trees of schedule delays on drainage pipeline activity

- Prediction of trends through Neural Networks
  9 Inputs

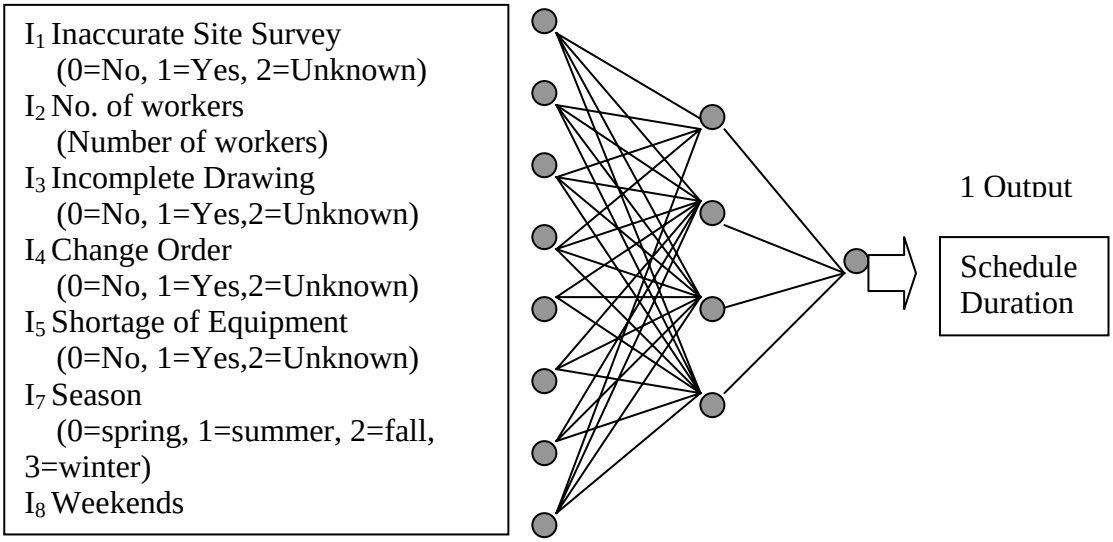

Figure 5. Description of NN inputs and output

Figure 5 shows the RMS data, with the nine input variables (Inaccurate Site Survey, Number of workers, Incomplete Drawing, Change Order, Shortage of Equipment, Duration, Season, Weekends, Rain/Snow) converted to numbers. The output value is the value of schedule delay for an activity.

The cycle is repeated for each case in the training set, with small adjustments being made in the weights after each case. When the entire training set has been processed, it is processed again. In this case study, the learning rate was 1 percent and the number of layers was 3, including 1 hidden layer.

As shown in Figure 6, the training error always decreases with an increase in the number of cycles. In contrast, the testing error does not have a continuously decreasing trend where a minimum value is found on the curve. Overfitting is the phenomenon where in most cases a network gets worse instead of better after a certain point during training. This is because such long training may make the network 'memorize' the training patterns, including all of their peculiarities. However, one is usually interested in the generalization of the network. Learning the peculiarities of the training set makes the generalization worse. The network should only learn the general structure of the examples.

Figure 6 shows the optimum point around 2,000 cycles where the training set error rate continues but test set error rate is bouncing back. Thus, it is noted in Figure 6 that the most appropriate number of cycle in the data set is 2,000 where training error rate is 10.56% and testing error rate 14.55%.

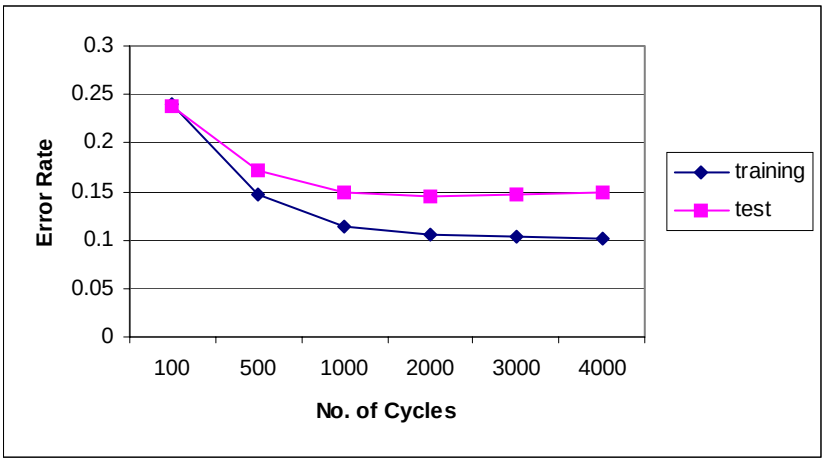

Figure 6. Comparison of error rates between training and testing sets

REFINEMENT PROCESS - COMPARISON BETWEEN TRADITIONAL AND KDD PREDICTIONS

After the analysis of results, the knowledge discovered needs to be examined by the experienced project manager in the field. This step also includes checking for and resolving potential conflicts with previously believed (or extracted) knowledge. Refinement could require redefining the data used in the discovery, a change in methodology, the user defining additional constraints on the mining algorithm, and so on since KDD process can involve significant iteration and may contain loops between any two steps.

According to the preliminary results of this case study, the main cause of schedule delays in the flooding control project at Fort Wayne was "Inaccurate site survey" rather than the weather related problems initially assumed by site managers. Discussions with site managers in the construction project confirmed the importance of equipment, such as ground penetration radar, to make the site surveys more accurate.

CONCLUSIONS

In this paper, the authors discussed an approach for discovering some useful knowledge from large amounts of data that were generated during a construction project. This paper introduced a knowledge discovery approach that is being developed for this real world application. This approach consists of four steps; (i) identification of problems, (ii) Data preparation, (iii) Data mining, (iv) data Analysis, and (v) Refinement process. The proposed approach helps to guide the analysis through the application of diverse discovery techniques. Such a methodological procedure will help us to address the complexity of the domain considered and therefore to optimize our chance to discover valuable knowledge.

During the knowledge discovery, one of the most important, time-consuming and difficult parts of KDD steps is data preparation. Domain knowledge and good understanding of the data is key to successful data preparation. Given the massive amounts of data that has become available nowadays, KDD can help in finding patterns and relationships in data that allow one to predict future results. By applying KDD to the large construction data, the project manager can have a better understanding of causal relationships in a project. With the use of very large database, this research utilizes KDD

technologies that reveal predictable patterns in construction data that was previously thought to be chaotic. The research of KDD application to large construction data is continuously being refined. Eventually, it will provide knowledge discovery model-building templates and wizards to guide novice model builders through the process of creating models based on their own construction data.

REFERENCES

Anand (1998), “Data Mining: Delving into the known”, Northern Knowledge Engineering Laboratory

Arditi & Gunaydin (1998), “Factors that affect Process Quality in the Life Cycle of Building Projects”, Journal of Construction Engineering and Management, Vol. 124, No. 3, May, 1998

Burati et. al. (1992), “Causes of Quality Deviations in Design and Construction”, Journal of Construction Engineering and Management, Vol. 118, No. 1, March 1992

Cabena et. al (1998), “Discovering Data Mining from Concept to Implementation”, International Technical Support Organization (IBM), Prentice Hall PTR

Davis et al. (1989), “Measuring Design and Construction Quality Costs”, Journal of Construction Engineering and Management, Vol. 115, No. 3., September 1989.

Fayyad et al. (1994), “Proceedings of KDD-94: the AAAI-94 Workshop on Knowledge Discovery in Databases”, AAAI Technical Report

Kohavi et. al. (1994), “Feature subset selection as search with probabilistic estimates”, in AAAI Fall Symposium on Relevance

Kohavi et. al. (1997), “Wrappers for Feature Subset Selection”, AIJ Special Issue

Liu et al (1998), “Feature Extraction Construction and Selection: A Data Mining Perspective”, Kluwer Academic Publishers

Peitgen (1994), "Chaos and Fractals", Jurgens and Saupe

Quinlan, Ross (1993), “C 4.5 Programs for Machine Learning”, Morgan Kaufmann Publishers

Rao, (1997), “Cost Risk Assessment in Environmental Remedation: Towards a Causal-Chain Approach”, Ph.D. thesis, University of Illinois at U-C

Rumelhart, D.E., (1986), "Learning internal representations by error propagation", In Parallel Distributed Processing, The MIT Press, Cambridge, MA

Rumelhart, D.E., (1994), "The Basic Ideas in Neural Networks", Communications of the ACM

Sanders et al. (1993), “Managing implementation of change”, Journal of Construction Engineering and Management.

Santos et al (1999), "Potential of Poka-Yoke Devices to Reduce Variability in Construction", Seventh Annual Conference of the International Group for Lean Construction (IGLC-7)

Suchman, E. A., Evaluative Research, Russell Sage Foundation, New York, 1967

USA CERL (1998), “Resident management System (RMS)”, US CERL (Construction Research Engineering Lab.)

Womack, James., Jones, Daniel T., (1990), “The machine that Changed the World”, Rawson, New York. 323 p

Yates (1993), “Construction Decision Support System for Delay Analysis”, J. Constr. Engrg. And Mgmt., ASCE

Womack et al (1990), "The machine that changed the world", Rawson Associates, Simon & Schuster Inc., New York, New York