The Information Institute

JIST 2(2) 2005

Journal of Information Science and Technology

www.jist.info

Building Knowledge Repositories with Neural Networks and Orthogonal Designed Bases of Expert Holistic Judgments

Dat-Dao Nguyen

Department of Accounting and Information Systems
California State University,
Northridge CA 91330-8372

Dennis S. Kira

Department of Decision Sciences and MIS
Concordia University,
Montreal-Canada H3G 1M8

Abstract

This paper proposes and illustrates a framework of building knowledge repositories with Neural Expert Systems (NES) to overcome difficulties often encountered by conventional experts systems technologies. The proposed system uses Neural Networks (NN) to capture decision patterns / production rules in a set of holistic judgments provided by experts on a minimal sample of cases taken from a problem domain. A NN learns tacit knowledge from implicit relationships between decision attributes and outcome. Holistic judgments overcome the difficulty of explaining explicitly production rules and heuristics of the experts. An orthogonal plan defines a minimal sample to acquire the initial training set for NN and to alleviate experts from the cognitive burden of specifying a complete set of production rules. Starting from this initial knowledge base, counter-examples given by experts when the system is in production are added to subsequent training. The knowledge base of a NES and consequently the knowledge repository will grow over time with additional patterns learning from its own production and expert opinions.

Keywords: Artificial Intelligence, Decision Support Systems, Expert Systems, Holistic Judgments, Knowledge Acquisition, Knowledge Repository, Main-effect Orthogonal Plan, Multi-criteria Decision Making, Neural Networks.

Introduction

Expert judgment is the qualitative or quantitative response of a knowledgeable person to a technical issue in a decision problem (Turban et al., 2004). Since expertise is scarce and not always available, Expert Systems (ES) seek to capture/transfer expertise from human experts to computer-based knowledge repositories and then on to other human non-experts (Metaxiotis & Psarras, 2003; Liao, 2004). To map expert knowledge into the current state of ES technology, know-how of experts is required to be explainable in a detail manner (Cowan, 2001). However, literature of cognitive psychology points out that most experts cannot explicitly express how decisions have been reached (Ericsson & Simon, 1980; Anderson, 2000). Literature on knowledge management also indicates that tacit knowledge on problem domain is most important/substantial to organizations (Nonaka , 1994; Kakabadse et al., 2001; Smith, 2001). Consequently, knowledge acquisition with conventional ES has to deal with seemingly incomplete and/or inconsistent expert information when exploring and capturing this intellectual resource.

Addressing these issues, this paper proposes an effective framework to build knowledge repositories with Neural Expert System (NES) to overcome the difficulties of conventional ES technology. In this system, a Neural Network (NN) (Rumelhart et al., 1986) is used to capture production rules in expert holistic judgments on a specifically designed sample of decision patterns taken from a problem domain. Consequently, the system is able to learn tacit knowledge by mapping implicit relationships between decision attributes and outcome. Holistic judgments / assessments overcome the difficulty of experts in explaining explicitly the production rules of their heuristics (Slovic & Lichtenstein, 1971). Using orthogonal plans (Alderman, 1962) – an experimental design technique –, the sample of exemplary decision patterns will be minimal and consequently alleviate the cognitive burden of experts in preference assessments. The knowledge base of a NES will grow over time as the system learns new patterns in the same way human experts broaden their knowledge.

The paper is organized as follows. Section 2 reviews difficulties of knowledge engineering with conventional ES in capturing implicit production rules and tacit knowledge. Section 3 proposes a framework for building NES in which neural network is the engine to learn patterns of holistic judgments and orthogonal designed sample of production rules is the initial knowledge base. Section 4 presents some illustrations of the framework. The paper concludes with some directions for future research.

Difficulties of Conventional Expert Systems in Knowledge Engineering

From a cognitive perspective, knowledge of domain experts typically contains of facts, collected experience, judgmental rules, and procedural issues on association, analogies, hypotheses, reasoning and intelligent choices (Chorafas, 1990; Anderson, 2000). Through a formal channel, experts accumulate a store of *facts* or factual information in their long-term memory to develop production *rules* or pattern-action *schemata* for future problem solving (Larkin et al., 1980).

In interactions with the environment, experts build internal, mental models of themselves and things they are interacting with (Norman, 1983). These *mental models* are definite representations embodying structural and/or functional properties of environmental entities. They provide predictive and explanatory power for understanding the interaction of factors in a problem domain and performing a complex task (Norman, 1983). However, it is known that human mental models are often messy, incomplete, inconsistent, imprecisely defined, and containing erroneous concepts (Norman, 1983).

The ability of experts to rapidly access knowledge also depends on the *indexing of knowledge* in their knowledge stores (Anderson, 1983). This is the acquisition of production rules to associate task-relevant knowledge with each pattern. The condition part of a condition-action rule represents the indexing pattern, which once satisfied will evoke the action (Larkin, 1981; Anderson, 1983). Expert skills are also stored in some unanalyzable procedures or macros. But evidence suggests that experts cannot always verbalize this *proceduralization of knowledge* (Brown & Burton, 1978).

The mental process of an expert is generally too complex to be described through algorithms (Chorafas, 1990). In many cases, the process is automated, implying a run to completion automatically once initiated, but unavailable for introspection (Shiffrin & Schneider, 1977). It has been found that experts rely less on deductive reasoning and more on pattern recognition in problem solving (Jeffrey & Ossario, 1976; Graff, 1979; Slatter; 1987). When retrieving facts from the indexing knowledge, perceptual patterns in the current state of problem solution automatically trigger the activation of a relevant piece of knowledge (Smith et al., 1978; Chase & Ericsson, 1982).

Knowledge Engineering or the knowledge acquisition process for ES attempts to obtain necessary information to represent the knowledge by production rules in the form of cause/effect, situation/action, if-then-else (Chorafas, 1990; Hoffman et al., 1995). The process requires codifying and making rules and/or procedures explicit, so that the captured knowledge could be programmable in a computer-based ES for future problem solving (Cowan, 2001). It reveals personal experience, historical information, inference strategies, reasoning processes, personal knowledge, personal heuristics, personal expertise, problem-solving procedures, justifications, and explanations from the experts (Martin & Oxman, 1988). But it may be very difficult for the experts to express introspectively about their decision process and inner experiences, especially when experiences are made up of intuition (Smith et al., 1978; Chase & Ericsson, 1982). Personal belief structures are not always available for inspection (Norman, 1983). Experts are often unaware of the detailed process that they have used to arrive at a conclusion (Nisbett & Wilson, 1977; Ericsson & Simon, 1980). Verbal protocols may be informative but incomplete. The technique may even yield erroneous information since people may state their beliefs but actually believe in another thing and frequently act in quite a different manner (Berry & Broadbent, 1984).

In spite of these difficulties in expressing production rules and decision-making process, experts can provide examples of problems and solutions reflecting their conceptualization of a domain (Michie, 1984). Hidden behind a judgment or holistic preference of problem alternatives, there is always an implicit logic on the association between input-output patterns (Slovic & Lichtenstein, 1971). Consequently, the machine learning approach could be an effective means to recognize and reveal these decision patterns / production rules. It offers the possibility of deducing new knowledge by listing all factors that influence a decision, without understanding in detail their impacts, and inducing a rule that works successfully (Turban et al., 2004).

A disadvantage of machine learning techniques is in the requirement of a database containing sufficiently documented cases structured around expert knowledge on a problem domain (Nieddu & Patrizi, 2000). Such a sample could be too large and consequently demand considerable cognitive effort from experts in the preference assessment process (Anderson, 2000). In addition, algorithms of machine learning techniques have used statistical methods such as decision tree technique, discriminant analysis, step-wise regression, principal component analysis, factor analysis, etc … (Berry & Linoff, 1997). It has been noted that traditional methods cannot handle the nonlinear relationship without imposing strong assumptions / restrictions on the behavior of data and functional models (Refenes, 1995; Nieddu & Patrizi, 2000).

In view of limitations of current techniques, this paper proposes an effective method of using neural networks to capture tacit knowledge from decision patterns underlying in expert holistic judgments without making unrealistic assumptions on data and functional models. The proposed method also relieves the heavy cognitive burden on experts with the assessment of a minimal sample of production rules in the knowledge engineering process.

Building Knowledge Repositories with Neural Networks

Neural Network (NN) has emerged as a prominent machine learning technique to map nonlinear relationships and underlying discriminant patterns between decision attributes and outcomes in a problem domain without requiring elaborated models or *a priori* assumptions on the data (Rumelhart et al., 1986). The technique is especially useful to capture tacit knowledge from input-output patterns in heuristics / holistic judgments provided by experts (Slovic & Lichtenstein, 1971).

Previous works have investigated the integration of NN and ES in Connectionist Expert Systems (Gallant, 1988) or Expert Networks (Medsker, 1995). But the implementation of these ES is hindered by the availability of an exemplary set of patterns and the cognitive effort of experts in providing judgment on these examples (Hoffman et al., 1995; Nieddu & Patrizi, 2000). Most important, these works did not consider the trade-off in multi-criteria decision making, which transcends all business problems but could not be expressed explicitly by the experts (Refenes, 1995).

In the following, the ability of NN in learning production rules and nonlinear behavior from expert holistic judgments is reviewed. Then the implementation of an orthogonal plan to design a minimal set of exemplary cases of decision patterns for NN training is discussed. After that a framework for building NES is presented.

Neural Networks to Learn Tacit Knowledge in Holistic Judgments

A neural network contains processing/computing units called neurons (or nodes). These nodes are arranged in layers in which a node of one layer has a weighted connection to each node of next layer in a particular configuration. The computational process is as follows. A node, as a processing unit, receives inputs from a number of other nodes or from an external stimulus. A weighted sum of these inputs constitutes the argument to an activation function or transfer function. Resulting value of the activation function is the node output that will be propagated along the weighted connections to other nodes. In a neural network, all knowledge is encoded in its interconnection weights. A weight represents the strength of the association among connected features, concepts, propositions, or events presented to the network (Refenes, 1995). A neural network learns by means of weight adaptation with a particular learning rule in the training phase.

A network topology consists of nodes as autonomous processing units connected by directed arcs. A neural network is classified either as feedforward if it does not contain directed cycles or recurrent if it does. Recurrent network is suitable for temporal events in which a current output is affected by the previous one. Each arc in a network topology has a numerical weight $w_{i,j}$ specifying the influence of node u_j on node u_i . Every node, other than input nodes, computes its new activation u_i as a function of the weighted sum of inputs directed to it from other nodes:

$$S_i = \mathrm{S}^{n}_{j=1} w_{i,j} u_j \qquad (1)$$

$$u_i = f(S_i) \qquad (2)$$

The activation function, $f(.)$, is usually nonlinear, bounded and piecewise differentiable such as the sigmoid function,

$$f(x) = 1/(1 + e^{-x}) \qquad (3)$$

Most NN applications have used 3-layer networks consisting of one input, one hidden, and one output layer. Such a NN produces a response, which is the superposition of n sigmoid functions, where n is the number of hidden nodes, to map a complex function. In some cases, more hidden layers are necessary to map a higher order function. From a mathematical perspective, it has been proved that a NN is a universal function approximator (Hassoun, 1995) and standard multi-layer feedforward networks using arbitrary transfer functions can approximate any continuous as well as discontinuous functions (Cybenko, 1989; Hornik et al., 1989).

Since knowledge possessed by a neural network is encoded in its interconnection weights, a training algorithm seeks to find an appropriate set of weights that produces the desired network output. A network can learn in a supervised mode (to map implicit input-output relationships with expert opinions as benchmarks) or in a self-organized mode (in Data Mining and Knowledge Discovery for new patterns). To train a neural network in a supervised mode, one presents it with a set of input patterns and associated output patterns (or targets). The choice of training algorithm and network configuration depends on data and therefore it becomes specific to the problem.

Among available algorithms, the backpropagation (BP) algorithm with a sigmoid transfer function is commonly used in a feedforward network (Rumelhart et al., 1995). The BP algorithm applies the mean squared error and gradient descent approach for the convergence between network outputs and desired outputs. The algorithm begins by randomly assigning small initial weights to break the symmetry so that various intermediate nodes can take roles. Then, one chooses a training example and computes the gradient with respect to it. The computation involves a forward pass over the network to compute node activations, following by a backward pass to compute gradients. Once the gradient is determined, a small step is taken to update the weights. The process continues until the algorithm converges (Lippmann, 1987; Gallant, 1993).

Theoretically, NN can learn from historical data and generalize over the new cases. NN can learn the associations between groups, elements of the problem domain. NN can approximate a functional relationship by utilizing a set of input-output patterns and learn probabilities and statistical distributions from the data themselves (Hassoun, 1995).

Practically, NN can offer the advantage of computer execution speed. The ability to learn from cases and train the system with new data rather than to rewrite programs may be more cost effective and even more convenient than conventional ES when frequent updates to emergent information become necessary (Medsker, 1994). In applications where rules are unknown, NN may be able to represent those rules implicitly as stored connection weights (Refenes, 1995).

A caution is that a NN cannot predict well patterns that it has not learned. As a result, one needs a large training set to capture all possible production rules taken from a multi-dimensional problem space. But limitation of human experts in information processing makes the assessment on a large number of cases impractical (Anderson, 2000). A randomly selected sample of fewer cases may not provide necessary information on problem domain (Nieddu & Patrizi, 2000) . A viable alternative is to assess expert judgments and train NN with representative examples of production rules in the problem domain. Using an *orthogonal main-effect plan*, one can design a minimal sample set of input-output patterns to train the networks and build the initial knowledge base for an integrated NES.

Orthogonal Main-Effect Plan to Design Initial Knowledge Base of Neural Network

To systematically capture preference on alternatives in a problem domain with conventional ES, one has to investigate effects of every combination of factor/criteria levels. Let n be the number of factors/criteria of the alternatives and s be the levels of these factor/criteria. If the problem domain includes factors with the same number of levels (a symmetrical factorial experiment), then the number of combinations to be assessed is s^n. If the factors have different number of levels (an asymmetrical factorial experiment), for example, n_1 factors have s_1 levels each, n_2 factors have s_2 levels each, then the number of combinations to be assessed is $s_1^{n1}s_2^{n2}$. Obviously, when the problem domain is large, the number of combinations to be assessed would impose a heavy cognitive burden in information processing on experts. To alleviate this cognitive burden, one can use experimental design to obtain a smaller sample of only a portion/fraction of all combinations of factors/criteria in the problem domain. Such sampling scheme and experimental designs are called *fractional factorial experiments* (Kirk, 1995; Berger & Maurer, 2002). *Orthogonal* matrices/plans can capture a body of expert judgments / decision patterns on decision factors and their interactions in "judgment spaces" (Jeffrey & Ossario, 1976; Graff, 1979).

One of these experimental design scheme providing smallest samples for preference assessment is the *orthogonal main-effect plan* (Addelman, 1962; Barron & Pearson, 1979). This experimental design permits assessing effects of several factors without considering every combination of factor levels (Addelman, 1962). Consider a symmetrical factorial experiment, a $(s^n - 1)/(s - 1)$ factorial design plan can represent alternatives of a problem space having n factors, each with s levels and their generalized interactions (Addelman, 1962). For an asymmetrical factorial experiment, the orthogonal main-effect plan can be constructed by collapsing factors occurring at s_1 levels to factors occurring at s_i levels by using a many-to-one correspondence of the set of s_1 levels to the set of s_i levels (Addelman, 1962).

It has been estimated that, by using the orthogonal main-effect design, a set of 49 holistic assessments of production rules can cover the dimensionality of decision problems having up to eight factors/criteria, with seven levels per factor/criterion (Barron & Pearson, 1979). For example, in a four-criterion problem with five levels per criterion, instead of making 5^4 or 625 assessments to define the production rules for a conventional ES, one needs only 25 assessments in an orthogonal plan to capture the main effects of the problem factors. Addelman (1962) provides a comprehensive catalogue of basic orthogonal plans and variations. Using this catalogue, one can capture effects in a problem domain having 2^7 treatment combinations with 8 trials (examples of alternatives) in *Basic Plan 1*, 3^4 treatment combinations with 9 trials in *Basic Plan 2*, 4^5 or 2^{15} treatment combinations with 16 trials in *Basic Plan 3*, etc… The plans specify which levels of each factor/criterion should be presented in each example of the sample set in order to capture the main effects of decision criteria. An orthogonal main-effect plan not only helps reducing the information processing burden to the experts, but also in defining the initial training set of production rules for a NES to acquire its knowledge base.

A Framework for Building Neural Expert Systems

The Neural Expert Systems (NES) approach proposed herein provides a unified framework by seeking to represent the mind with the rules and describing the function of the brain with the nonlinear / parallel distributed processing of a neural network (Dreyfus & Dreyfus, 1990). This system acquires its knowledge through learning by seeing examples of a problem domain, learning by doing when reproducing these examples, and learning by being told when getting feedback to improve its performance. Starting with a basic knowledge base, new knowledge will be accumulated when the system is exposed to new cases. This is appealing since the learning and knowledge acquisition process of a NES is similar to the way that human beings acquire knowledge (Michalski, 1987).

1. Select an appropriate orthogonal plan to define the minimal sample of exemplary decision alternatives in a problem domain.
2. Acquire holistic judgments of expert on these training examples to build the initial knowledge base.
3. Train the network to learn decision patterns from the initial knowledge base.
4. Generalize the trained network over new examples.
5. On the opinion of the expert, add counter-examples to the training set
6. Retrain the network to learn new patterns and expand the knowledge base.

Table 1. Knowledge Acquisition Process of NES

The framework for building a NES is presented in Table 1. Within this framework, one starts with an appropriate orthogonal plan selected from the Addelman catalogue (Addelman, 1962) to define a set of basic examples from a problem domain. As experts may not explain in details their decision making process, holistic assessments of these examples should overcome this difficulty of expressing explicitly the rules and procedures to arrive at a decision. These holistic judgments constitute a training set for neural network to acquire an initial knowledge base on the problem domain.

Starting with a fully connected network architecture, one obtains an initial set of weighted connections representing the basic knowledge base of NES. Then node interconnections can be pruned to achieve an optimal network configuration in which node connections and weights represent production rules or influences of input patterns on output. A set of learned rules could be pre-wired in a NES with the presence/absence of directed arcs from one node to others or the setting of some predefined weight values for some arcs in node interconnections.

A trained network will be used to make generalization over new cases. Experts will evaluate network performance on these cases. Any wrong predictions will constitute counter-examples to be added to the initial training set. Then the network will be retrained to learn emerging patterns in these counter-examples. This process of learning by doing will continue as needed to acquire a robust knowledge base for NES. Over time, a NES will learn and enrich its knowledge base with new patterns of counter-examples.

The advantage of NES is in its handling of nonlinearity, multicolinearity and trade-offs among factual data representing in a minimal sample. The nonlinearity rends traditional statistical models insufficient to capture the relationship of data. The multicollinearity and trade-offs among data make the extraction of explicit rules become difficult for traditional statistical methods and conventional ES. In contrast, as a universal function approximator, a NN can learn any relationship in data without imposing any *a priori* assumptions on their behaviors and functional models. The logic hidden behind holistic judgments and implicit production rules is captured in the connection weights of NN. Although it is not easy to explain the interrelationship, one can still assess the influence of one problem factor to the others in arriving at a satisfactory final output.

A caution is that a trained and validated network does not always guarantee a well generalization. However, this is the same drawback of traditional statistical methods when they make inference on a point outside the range of sample data. In fact, to some extent, human experts also experience the same difficulty with new decision problems in which patterns are different from those of their mental representations (Anderson, 2000).

Illustrations

This section illustrates the implementation of NES in two scenarios. These scenarios represent common types of decision-making problems, in which one either makes a choice based on qualitative rankings and/or quantitative judgments of multi-criteria alternatives. The first scenario is to assess a binary decision whereas the second one is to approximate preference scores of many alternatives.

Fundamental Analysis of Accounting Data

In the following, the implementation of a NES for fundamental analysis of accounting data is presented. Based on suggestions in finance literature, the Asset Management module of this NES has four criteria: Average Collection Period, Inventory Turnover, Fixed Asset Turnover, Total Asset Turnover (Gibson, 1992; Helfert 1994; Block & Hirt, 2004). Each criterion has a binary value [Favorable, Unfavorable] based on a predetermined threshold. From related literature and opinion of a finance expert, these threshold values are specified in the following rules.

If Average Collection Period LE 6.4 then Favorable, else Unfavorable.

If Inventory Turnover GE 13.2 then Favorable, else Unfavorable.

If Fixed Asset Turnover GE 5.3 then Favorable, else Unfavorable.

If Total Asset Turnover GE 3.2 then Favorable, else Unfavorable.

In a composite rule for assessing performance of asset management, there exist tradeoffs among criteria that make their representation in an appropriate rule set for conventional ES extremely difficult. Taking into account these tradeoffs in their assessment, experts might provide judgments that seem to be contradiction and/or inconsistent with the formal logic.

This decision problem has four criteria with two levels of value [Favorable, Unfavorable] per criterion. To capture the rule set in such a problem domain with a conventional ES, one needs to ask a finance expert to make judgments on a set of 2^4 or 16 production rules and inputs them into the knowledge base

In a small problem domain like this, one may not have any difficulty in capturing a complete set of production rules. However, as the number of criteria and/or levels becomes larger, the rule set of ES will grow exponentially and will demand greater cognitive effort from an expert. This burden could be alleviated with the implementation of an orthogonal plan. The experiment reported herein demonstrates that even with a subset of production rules selected effectively with the orthogonal design, a NES could still arrive at an accurate suggestion.

An orthogonal plan in the Addelman catalogue (Addelman, 1962) was used to define a minimal training set for this problem (Table 2). This training set contains 8 cases from the problem domain

	Decision Patterns	Expected Outputs
Training Set		
	1111	1
	1001	1
	1110	1
	0100	0
	0011	1
	0111	1
	1010	1
	1101	1
Test Set 1		
	1011	1
	0001	0
	1100	1
Test Set 2		
	1000	0
	0110	1
	0010	0

Table 2. Production Rules for Asset Management Module of NES

The second case (1001) with an output of (1) from expert judgment is interpreted as follows.

"IF Average Collection Period is Favorable,
AND Inventory Turnover is Unfavorable,
AND Fixed Asset Turnover is Unfavorable,
AND Total Asset Turnover Favorable,
THEN Asset Management is Favorable."

With expert judgment available on each case in the initial rule sets, one trained a NN to acquire an initial knowledge base. To test the performance of this NES, one used the NN knowledge base to generalize on two sets of out-of-sample cases, namely (1100, 1011, 0001) and (1000, 0110, 0010). The same finance expert was asked to evaluate the network output. If there was any disagreement between network output and expert judgment, this counter-example with the correct response would be added to the initial training set and the network was retrained to learn this new pattern.

The NN in this scenario was built with four input nodes, one for each criterion, and one output node. Several 3-layer networks were experimented with the number of hidden nodes varying from 0 to 3. These networks used the BP algorithm with a sigmoid transfer function, a learning rate of 1, a momentum of .90. The training and testing tolerance were set at .10 and .30 respectively.

The simplest configuration was found to be the one with no hidden layer. This NN used a sigmoid transfer function for the output node. The learning algorithm converged at 446 training epochs. On the first test, the network assessed correctly the case (1011) as favorable and (0001) as unfavorable. However, it made an error on the case (1100) by predicting unfavorable instead of favorable. This counter-example was added to the training set and the network was retrained. In the second run, the learning algorithm converged at 506 epochs. Testing on the second set with new knowledge, the network assessed correctly (1000) as unfavorable and (0110) as favorable. But this time, it made an error on the case (0010) by predicting favorable instead of unfavorable. In this experiment, one notes that once the new pattern of the counter-example was added to the initial knowledge base, the weighted connection representing the influence of input on output was changed. This indicates that the network has adapted itself to capture the new knowledge. One also notes that if a linear transfer function (linear regression) was used, the network did not converge even after 15000 training epochs. This indicates that the traditional linear model cannot capture the underlying complex functional relationship in data.

Other network configurations with one and two nodes in the hidden layer were experimented. These networks made the same errors in generalization on the first and second tests. However, in these network configurations, the learning algorithms converged at 243 epochs and 215 epochs respectively in the first training and 264 and 222 epochs respectively in the second training after adding the counter-example. Although these networks converged faster, the use of these complex configurations (to approximate a high order function) may not be necessary in this scenario as a simpler network (to approximate a simpler function) can arrive at the same results.

Project Evaluation and Economic Appraisal

The decision problem reported herein relates to project evaluation and economic appraisal of new product proposals in a manufacturing company. A proposal is evaluated on five criteria: (i) Net Present Value (NPV) of cash flows generated over the next five years, (ii) initial capital investment requirement, (iii) market growth rate in the next five years, (iv) capability to market the new product, and (v) prospect of technical success. These criteria are suggested by related literatures (Souder, 1984; Meredith & Mantel, 2002) taking into account financial, marketing and production aspects of the proposed product. Some criteria are quantitative, whereas others are qualitative. In the appraisal process, quantitative data are converted into categories to avoid the difficulty of assessing and predicting value on a continuous and infinitesimal scale. For example, the flow of NPV is classified into five categories ranging from 1 to 5 million dollars. Similarly, the amount of initial investment is classified into five categories ranging from .5 to 2.5 million dollars. The criteria concerning market growth rate, capability to market the new product and prospect of technical success have three levels each, namely, fair, good and very good. The problem domain is represented in Table 3.

NPV of Cash Flow	$ [1.0	2.0	3.0	4.0	5.0]	millions
Initial Investment	$ [2.5	2.0	1.5	1.0	0.5]	millions
Market Growth Rate	[fair	good		very-good]		
Capability to Market	[fair	good		very-good]		
Prospect of Technical Success	[fair	good		very-good]		

Table 3. Project Evaluation and Economic Appraisal: The Problem Domain

Before submitting a proposal, the related department had estimated values for each project criterion. In the appraisal process, there existed tradeoffs among criteria and no single criterion absolutely dominates any others. The experts rated each proposal on a scale ranging from 0 to 100. In this study, appraisals were taken from a senior manager, a senior engineer and a chartered accountant. These experts were indicated as Expert A, Expert B, and Expert C respectively.

To build a knowledge base with a conventional ES, a complete set of $5^2 3^3$ or 675 production rules would be needed for assessment purpose in this decision problem. In contrast, within the framework proposed herein, an orthogonal plan from the Addelman catalogue (Addelman, 1962) was used to define a sample of only 24 examples of production rules from the problem domain. Then each expert was asked to provide preference judgment for each example. For instance, the preference of Expert A for the first exemplary pattern (1 2 2 3 3) with the score of 50 from Expert A in Table 4 is interpreted in the following production rule.

"IF NPV is 1 million dollars
AND Initial Investment is 1 million dollars
AND Market Growth Rate is Good
AND Capability to Market is Very Good
AND Prospect of Technical Success is Very Good
THEN the Proposal is rated 50 /100."

In this manner, the sample constituted a set of production rules implicitly expressed by each expert. Using this set, one trained NN to acquire an initial knowledge base. Expert judgments were also assessed on another two test sets of five out-of-sample examples for validating the system performance. On expert opinion, any disagreement between network prediction and expert judgment in the test sets would constitute a counter-example to be added into initial training set. The network was then retrained to learn the new pattern and make generalization in future. Consequently, the knowledge base of the system would grow over time with the learning of new patterns in counter-examples.

In this experiment, NN for each expert was configured with five input nodes, one for each criterion, five hidden nodes and one output node. These networks used a backpropagation algorithm with sigmoid transfer functions, a learning rate of 1, and a momentum of .9. The training and testing tolerance were set at .1 and .3 respectively.

	Expert A		**Expert B**	
Training Set	*Score*	*Prediction*	*Score*	*Prediction*
1 2 2 3 3	50	54.63	45	47.64
1 3 3 1 2	40	39.50	46	48.07
1 4 3 2 1	45	44.90	54	53.86
1 5 1 3 3	60	60.49	65	62.25
2 1 2 2 2	35	33.43	42	44.89
2 2 3 3 1	50	49.79	52	51.02
2 3 3 1 3	55	55.08	57	56.56
2 4 1 3 1	40	38.26	58	57.88
2 5 1 1 3	45	44.46	63	64.50
3 1 3 3 3	70	69.89	58	56.85
3 2 3 1 1	35	34.78	52	51.85
3 3 1 2 3	50	49.48	62	61.35
3 4 1 3 2	55	55.81	69	69.81
3 5 2 1 1	40	40.39	70	70.78
4 1 3 3 3	75	73.78	65	65.46
4 2 1 1 2	30	29.72	55	53.62
4 3 1 3 1	45	43.61	66	66.33
4 4 2 1 3	60	60.88	76	76.05
4 5 3 2 1	65	68.24	83	82.35
5 1 1 1 1	20	24.21	52	49.95
5 2 1 2 3	55	54.31	68	69.03
5 3 2 3 1	60	61.66	75	77.07
5 4 3 1 3	75	73.99	85	82.91
5 5 3 3 2	90	82.35	93	87.42
Test Set 1				
3 3 1 2 1	30	29.83	55	54.75
5 1 3 1 1	40	38.39	58	58.26
1 5 2 3 3	70	69.51	68	65.09
4 4 2 2 2	60	62.35	76	77.24
1 4 3 3 2	65	65.45	62	58.44

(Contd...)

Test Set 2				
3 1 2 2 2	40	38.16	48	48.65
3 4 1 2 2	45	44.49	63	66.28
1 4 3 3 3	75	72.76	65	60.48
2 2 2 2 3	50	49.44	52	50.61
4 1 1 2 3	45	42.47	55	53.71

Table 4. Learning and Predicting Preference of Experts A and B.

For Expert A, the network converged after 119 training epochs with a Root Mean Square of Errors (RMSE) of .0304. In generalization over two test sets, it made RMSE of .0170 and .02275 respectively. For Expert B, the network converged after 104 training epochs with an RMSE of .0333. In generalization over two test sets, it made RMSE of .0381 and .0473 respectively. On the opinions of these two experts, the system had provided satisfactory generalization on both test sets.

First Run			**Second Run**		
Training Set	*Score*	*Prediction*	*Training Set*	*Score*	*Prediction*
1 2 2 3 3	30	26.95729	1 2 2 3 3	30	27.86282
1 3 3 1 2	40	38.94524	1 3 3 1 2	40	37.15381
1 4 3 2 1	60	59.76158	1 4 3 2 1	60	60.95514
1 5 1 3 3	50	50.36586	1 5 1 3 3	50	50.32877
2 1 2 2 2	30	32.01082	2 1 2 2 2	30	34.2081
2 2 3 3 1	60	60.53619	2 2 3 3 1	60	60.18925
2 3 3 1 3	50	51.25394	2 3 3 1 3	50	52.09837
2 4 1 3 1	60	59.45174	2 4 1 3 1	60	59.10262
2 5 1 1 3	70	69.31223	2 5 1 1 3	70	69.52824
3 1 3 3 3	95	87.93129	3 1 3 3 3	95	88.05566
3 2 3 1 1	95	94.61041	3 2 3 1 1	95	92.54405
3 3 1 2 3	80	81.79112	3 3 1 2 3	80	80.6783
3 4 1 3 2	90	89.84055	3 4 1 3 2	90	92.66843
3 5 2 1 1	90	96.23819	3 5 2 1 1	90	96.90807
4 1 3 3 3	80	84.30698	4 1 3 3 3	80	83.20506
4 2 1 1 2	70	68.95001	4 2 1 1 2	70	67.50552
4 3 1 3 1	70	70.09775	4 3 1 3 1	70	69.33841
4 4 2 1 3	90	88.72336	4 4 2 1 3	90	88.78882
4 5 3 2 1	95	96.38002	4 5 3 2 1	95	97.11317
5 1 1 1 1	60	60.36818	5 1 1 1 1	60	60.69984
5 2 1 2 3	60	59.5041	5 2 1 2 3	60	61.01624
5 3 2 3 1	80	79.36691	5 3 2 3 1	80	79.62003
5 4 3 1 3	90	89.9671	5 4 3 1 3	90	90.27477
5 5 3 3 2	95	95.8607	5 5 3 3 2	95	97.06953
			3 3 1 2 1	70	70.63234
			5 1 3 1 1	65	65.18605

Test Set 1			*Test Set 2*		
3 3 1 2 1	70	81.70384	3 1 2 2 2	50	63.24843
5 1 3 1 1	65	79.42801	3 4 1 2 2	60	93.34921
1 5 2 3 3	70	63.60191	1 4 3 3 3	60	59.30336
4 4 2 2 2	95	94.17838	2 2 2 2 3	40	49.93819
1 4 3 3 2	60	59.23135	4 1 1 2 3	60	55.38448

Table 5. Learning and Predicting Preference of Expert C

The case of Expert C is illustrated in Table 5. In this case, the network converged after 2425 training epochs, with an RMSE of .0334. On the first test set, it made two wrong predictions with an estimate of 81.70 for the example having a preference score of 70 and an estimate of 79.42 for the example having a preference score of 65. These two counter-examples were added into the training set and the network was retrained to learn the new patterns. In the second run, the network converged at 1769 epochs with an RMSE of .0369. On the second test set, it made only one wrong prediction with an estimate of 93.34 for the example having a preference score of 60. The network training for this case took longer time to learn the decision patterns and production rules of Expert C. However, the system's knowledge base and predictability for the preference of Expert C increased over time as it had learned more about decision patterns of this expert.

These above illustrations show that a NES performs well with a knowledge base acquired from holistic judgments on a sample containing only a subset of all production rules in the problem domain. The advantage becomes more apparent when one deals with decision problems having higher dimensionality. In such situations, the use of orthogonal main-effect plan provides a minimal sample of cases from the domain. Consequently, less cognitive efforts required as experts would make judgments on fewer examples. In addition, the holistic assessment would help experts overcome the difficulty in expressing explicitly the decision rules of their heuristics.

Conclusions

To address the concern on tacit knowledge in knowledge management and the difficulties of conventional ES technologies in capturing this intellectual resource, this paper has proposed and illustrated an effective framework to integrate NN and ES in a Neural Expert System (NES). The production rule set of NES is captured by training a NN to learn patterns in a set of holistic judgments provided by experts on a minimal sample of alternatives taken from a problem domain. Depending on the dimensionality of the problem, an appropriate orthogonal plan is selected to define a basic training set for the neural network and therefore to build up an initial knowledge base for the NES. This basic training set is designed to reduce the cognitive burden of experts in expressing production rules. It does not however diminish the capacity of representing the expert knowledge. Holistic assessments would help experts overcome the difficulty in expressing explicitly their decision rules and their heuristics. The knowledge base of NES will grow over time by learning new patterns from counter-examples, identified from expert opinions, in subsequent generalization.

From this point, there are many directions in which the NES could be further developed. In this paper, the engine of a NES is a fully-connected NN using a BP learning algorithm. Actually, depending on the complexity of a problem domain, other network configurations and learning algorithms could be implemented for knowledge discovery and acquisition. Genetic Algorithms (GA) (Holland, 1975; Goldberg, 1989) could be used to select an optimal network configuration in terms of number of hidden nodes and layers as well as appropriate transfer function.

In addition, multiple expert judgments could be integrated to enhance the performance of the system. Elsewhere, we have combined multiple expert judgments in a composite training set in a group decision context. We are also experimenting the implementation of "mixture- of-experts of neural networks" (Jordan & Jacobs, 1995), in which many neural networks are functioning concurrently and each of which representing a knowledge base captured from individual expert. The knowledge base of such a mixture-of-experts network will represent a knowledge repository of production rules provided by multiple decision makers in a multiple criteria decision-making context.

References

Addelman S. (1962). Orthogonal Main-Effect Plans for Asymmetrical Factorial Experiments. *Technometrics*, (4)1, 21-46.

Anderson J.R. (2000). *Cognitive Psychology and Its Implications (5th Ed.)*. New York: Worth Publishers.

Anderson J.R. (1983). Acquisition of Proof Skills in Geometry. In J.G. Carbonell, R. Michalski and T. Michell (Eds.), *Machine Learning, An Artificial Intelligence Approach*, San Francisco: Tioga

Barron F.H. & Person H.B. (1979). Assessment of Multiplicative Utility Functions via Holistic Judgments. *Organizational Behavior and Human Preference*, 24, 147-166.

Berger P.D. & Maurer R.E (2002). *Experimental Design.* Pacific Grove, CA: Duxbury.

Berry D.C. & Broadbent D.E. (1984). On The Relationship Between Task Performance and Associated Verbalizable Knowledge. *Quarterly Journal of Experimental Psychology*, 36A, 209-231.

Berry M. & Linoff G. (1997) *Data Mining Techniques*. New York: J. Wiley.

Block S. & Hirt G. (2004). *Foundations of Financial Management* (11th Ed.) Boston: McGraw-Hill.

Brown J.S. & Burton R.R. (1978). Diagnostic Models for Procedural Bugs in Basic Mathematic Skills. *Cognitive Science*, 2, 155-192.

Chase W.G. & Ericsson K.A. (1982). Skill and Working Memory. In G.H. Bower (Ed.), *The Psychology of Learning and Motivation*, Vol.16. New York: Academic Press.

Chorafas D. (1990). *Knowledge Engineering*. New York: Van Nostrand Reinhold.

Cowan R. (2001) Expert Systems: Aspects of and Limitations to the Codifiability of Knowledge. *Research Policy*, 30, 1355-1372.

Cybenko G. (1989). Approximation by Superpositions of A Sigmoid Function. *Mathematics of Control, Signals, and Systems*, 2, 303-314.

Dreyfus H.L. & Dreyfus S.E. (1990). Making a Mind versus Modelling the Brain: Artificial Intelligence Back at a Branch-Point. In M.A. Boden (Ed.), *The Philosophy of Artificial Intelligence*. Oxford: Oxford University Press.

Ericsson K. & Simon H.A. (1980). Verbal Reports As Data. *Psychological Review*, 87, 215-251.

Jeffrey H.J. & Ossorio P.G. (1976). *A New Technique for Implementing Human Knowledge and Abilities on Computer*. Technical Report, Systems & Information Department, Vanderbilt University, Nashville, TN

Jordan M.T. & Jacobs R.A. (1995). Modular and Hierarchical Learning Systems. In M. Arbib (Ed.), *The Handbook of Brain Theory and Neural Networks*, Cambridge, MA: MIT Press.

Gallant S.I. (1993). *Neural Network Learning and Expert Systems*. Cambridge, MA: The MIT Press.

Gallant S.I (1988). Connectionist Expert Systems. *Communication of the ACM*, 31(2), 152-169.

Gibson C.H. (1992). *Financial Statement Analysis: Using Financial Accounting Information*. Cincinnati: South-Western Publishing Co.

Goldberg D.E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley: Massachusetts.

Graff M.G. (1979). KIBIZT: The Design of a Bridge-Bidding program Using Simulated Human Judgment. *Proceedings of the 17th Southeast Regional Conference*,(Orlando April 1979). New York: ACM Press.

Hassoun M.H.(1995). *Fundamentals of Artificial Neural Networks*. Cambridge, MA: The MIT Press.

Helfert E.A. (1994). *Techniques of Financial Analysis*. Homewood, IL: R.D.Irwin.

Hoffman R., Shadbolt N.R., Burton A.N. & Klein G. (1995). Eliciting Knowledge from Experts: A Methodological Analysis. *Organizational Behavior and Human Decision Process*, 62(2), 129-158.

Holland J. (1975). *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: The University of Michigan Press.

Hornik K., Stinchcombe M. & White H. (1989). Multi-Layer Feedforward Networks Are Universal Approximators. *Neural Networks*, 2, 359-366.

Kakabadse N.K., Kouzmin A. & Kakabadse A. (2001). From Tacit Knowledge to Knowledge Management: Leveraging Invisible Assets. *Knowledge and Process Management*, 8(3), 137-154.

Kirk R.E. (1995). *Experimental Design: Procedures for the Behavioral Sciences*. Pacific Grove, CA: Brooks/Cole Publishing Company.

Larkin J.H. (1981). Enriching Formal Knowledge: A Model for Learning to Solve Textbook Problems. In J.R. Anderson (Ed.), *Cognitive Skills and Their Acquisition*, Hillsdale, NJ: Erlbaum.

Larkin J.H., McDermott J., Simon D.P. & Simon H.A. (1980). Expert and Novice Performance in Solving Physics Problems. *Science*, 208, 1335-1342.

Liao S.H. (2004). Expert System Methodologies and Application – A Decade Review from 1995 to 2004. *Expert Systems with Applications*, 28, 93-103.

Lippmann R.P. (1987). An Introduction to Computing with Neural Nets. *IEEE ASSP Magazine*, 4(2), 4-22

Martin J. & Oxman S. (1988). *Building Expert Systems*. Englewood Cliffs, NJ: Prentice Hall.

Medsker L.R. (1995). *Hybrid Intelligent Systems*. Boston: Kluwer Academic Publishers.

Meredith J.R. & Mantel S.J. (2002). *Project Management: A Managerial Approach*. 5th edition. New York: J. Wiley.

Metaxiotis K. & Psarras J. (2003). Expert Systems in Business: Applications and Future Direction for the Operations Researchers. *Industrial Management & Data Systems* 103/5, 361-368.

Michalski R. (1987). Learning Strategies and Automated Knowledge Acquisition: An Overview. In L. Bolc (Ed.), *Computational Models of Learning*. Berlin: Springer-Verlag.

Michie D. (Ed.)(1984). *Introductory Readings in Expert Systems*. New York: Gordon & Breach.

Nieddu L. & Patrizi G. (2000) Formal Methods in Pattern Recognition: A Review. *European Journal of Operational Research*, 120, 459-495.

Nisbett R.E. & Wilson T.D. (1977). Telling More than We Can Know: Verbal Report on Mental Processes. *Psychological Review*, 84, 231-259.

Nonaka T. (1994). A Dynamic Theory of Organizational Knowledge Creation. *Organization Science*, 5(1), 14-37.

Norman D.A. (1983). Some Observations on Mental Models. In D. Gentner and A.S. Stevens (Eds.), *Mental Models*. Hillsdale, NJ: Erlbaum.

Refenes A.P. (Ed.) (1995). *Neural Networks in the Capital Market*. Chichester: John Wiley.

Rumelhart D.E., McClelland J.L. & The PDG Research Group (1986). *Parallel Distributed Processing, Explorations in the Microstructure of Cognition; Vol.1: Foundations*. Cambridge, MA: The MIT Press.

Rumelhart D.E., Durbin R., Golden R. &Chauvin Y. (1995). Backpropagation: the Basic Theory. In Y. Chauvin and D.E. Rumelhat (Eds.), *Backpropagation: Theory, Architectures, and Applications*. New Jersey: Lawrence Earlbaum.

Shiffrin R.M. & Schneider W. (1977). Controlled and Automatic Human Information Processing: II. Perceptual Learning, Automatic Attending and A General Theory. *Psychological Review*, 88, 127-190.

Slatter P.E. (1987). *Building Expert Systems: Cognitive Emulation*. Chichester: Ellis Horwood Limited.

Slovic P. & Lichtenstein S. (1971). Comparision of Bayesian and Regression Approaches to the Study of Information Processing in Judgment. *Organizational Behavior and Human Performance*, 6, 649-744.

Smith E.A. (2001). The Role of Tacit and Explicit Knowledge in the Work Place. *Journal of Knowledge Management*, 5(4), 311-321.

Smith E.E., Adams N. & Schorr D. (1978). Fact Retrieval and the Paradox of Interference. *Cognitive Psychology*, 10, 438-464.

Souder W.E. (1984). *Project Selection and Economic Appraisal*. New York: Van Nostrand Reinhold.

Turban E., Aronson J.E & Liang T.P (2004). *Decision Support Systems and Intelligent Systems, 7th Ed.* Upper Saddle River, NJ: Prentice Hall.