Zeitschrift: IABSE reports = Rapports AIPC = IVBH Berichte

Band: 68 (1993)

Artikel: Machine learing in blackboard system for steel structures

Autor: Aji, K.B. / Kalyanaraman, V.

DOI: https://doi.org/10.5169/seals-51849

Machine Learning in Blackboard System for Steel Structures

Apprentissage automatique dans la construction métallique

Maschinen-Lernen nach dem Schultafelsystem im Stahlbau

K.B. AJI
Research Scholar
IIT Madras
Madras, India

V. KALYANARAMAN
Professor
IIT Madras
Madras, India

K.B. Aji, born 1962, graduated from REC Calicut, Kerala. Currently he is a doctoral student and a Senior Project officer in a project on expert Systems for steel struct. in the Civil Eng. Dep. at IIT Madras.

V. Kalyanaraman, born 1947, got his Bachelor in Civil Eng. at IIT Madras and PhD at Cornell Univ. Currently he is Prof. at IIT Madras specialising in stability, steel struct. and CAD.

SUMMARY

Machine learning paradigms in the recent decade have made considerable strides in the area of Artificia Intelligence. Eventhough structural engineering domain is a fertile ground for using these paradigms to improve engineering process, the literature in this area are only a few. This paper describes appropriate machine learning strageties for implementation in an integrated engineering system for knowledge based engineering of steel structures.

RÉSUMÉ

Au cours de la dernière décennie, une évolution paradigmatique considérable a eu lieu dans la conception de l'apprentissage automatique relatif au domaine de l'intelligence artificielle. Bien que la technique de la construction soit un milieu fertile pour utiliser ces paradigmes en vue de perfectionner les processus d'études, la littérature sur ce sujet reste limitée. Cet article décrit des stratégies adéquates d'apprentissage automatique pour être appliquées dans un système d'études intégrées destiné à la construction métallique.

ZUSAMMENFASSUNG

In Gestalt des Maschinen-Lernens hat sich im letzten Jahrzehnt mit grossen Schritten ein Paradigmenwechsel auf dem Gebiet der künstlichen Intelligenz vollzogen. Obwohl der konstruktive Ingenieurbau sich für Verbesserungen im Entwurfsprozess durch derlei Konzepte anbietet, gibt es nur wenig Literatur darüber. Der Beitrag beschreibt geeignete Strategien des Maschinen-Lernens zur Implementierung in ein integriertes Entwurfssystem für Stahlbauten.

1. INTRODUCTION

Machine learning enables a system to perform the same task or a task drawn from the same population more efficiently and effectively the next time [1]. Objectives of research on machine learning may be one of the following: i) simulate and thereby understand and improve human learning process, ii) develop natural language processing capabilities to serve as interface between man and machine, iii) improve problem solving skills of computing and iv) enhance learning from discovery.

The motivation for research which forms the basis for this paper, is to improve the problem solving capability of computer aided engineering systems by machine learning. Understanding the role and the application of machine learning strategies would facilitate acquisition of new knowledge, efficient reorganization of the existing knowledge, faster and better solution, expansion of the problem solving capabilities, learning of control knowledge, simulation of creative problem solving, and efficient solution even under uncertain and incomplete problem specification.

This paper deals with the machine learning strategies for computer aided engineering of steel structures in an extended blackboard system developed in a project on knowledge based expert system for integrated engineering of steel structures. Initially, generic paradigms in AI for machine learning are briefly reviewed in order to introduce the state of the art. Subsequently literature on the application of machine learning techniques in the civil engineering domain are discussed. It is shown that both the Machine learning techniques and applications are yet to deal with the needs of a large engineering domain. Finally the opportunities for and issues in machine learning in the engineering of steel structures are discussed and appropriate learning strategies are evolved for such a system. The discussion is illustrated with a few examples.

2. MACHINE LEARNING

AI research on machine learning over the past few decades has led to four well accepted machine learning paradigms, namely, inductive learning, analytic learning, genetic algorithms (classifier systems) and connectionist learning methods [2].

2.1 Inductive Learning

Formulation of plausible general assertion that explain given facts and prediction of new facts based on these general assertions is induction. Induction is an essential component of human learning. We induce a concept from a series of observations of a process or a phenomenon. Thus inductive learning involves the formation of a concept from examples and counter examples. In general induction can be either a single-shot process based on initial training examples or an incremental one. Induction is by far the most widely studied paradigm [3,4,5]. Gennari et al. [6] have identified the common features in induction learning such as unsupervised learning, incremental learning, integrated with performance, top down classification and incremental hill climbing.

The programs based on induction can handle inputs represented in a specific manner, such as attribute value pairs. This requires large scale structuring of the knowledge and hence limits the scope of the learning task. Moreover the learning is highly empirical, which constrains the extent of knowledge that can be learnt.

2.2 Analytic Learning

Analytic learning methods are deductive in nature and use the past experience in problem solving to arrive at the solution. These methods are superior to inductive methods as they can provide explanation for the classification of instances. The major contributions to this paradigm are in the areas of analogical reasoning [7,8,9], case based reasoning [10,11] and explanation based learning [12,13].

Analogical reasoning [14] consists of transferring knowledge from past problem solving episodes to new problems that share significant aspects with corresponding old experience and using it to construct solutions to the new problems. Case based reasoning also involves drawing conclusions from problems solved in the past to use in new problems. This kind of reminding of old experience [10] in the form of explanation can be processed by an EBL mechanism to generate new solutions. Explanation based learning involves generalizing the explanation obtained from an instance. Thus EBL produces a description of a concept based on the domain theory, which explains a particular instance of that concept.

2.3 Classifier Learning

Classifier systems are massively parallel, message passing, rule based systems that learn through credit assignment and rule discovery [15,16]. The algorithm used for rule discovery is analogous to the biological mutation process and hence the name genetic algorithm is also used for these systems. The learning process is closely similar to the inductive mechanisms and the connectionist methods. Although the nature of learning is highly empirical, under complex environments characterized by noisy and incomplete data this methods offers a viable alternative for learning.

2.4 Connectionist Learning

Connectionist methods, (also known as neural networks) emulate the function of mammalian brain. Typically a neural network [17, 18] consists of three different layers namely, the input layer, the hidden layer and the output layer. Each layer consist of a group of processing elements characterised by their weights. These processing elements enable the network to map the internal representation of a problem by suitably modifying their weights to match the input-output patterns. A concept can be represented over the entire network (distributed representation) or represented at a local level (localised representation). Once the network is trained with sufficiently large number of examples, it can generate solution to new problems. This method is highly suitable for parallel processing and is promising for future computing requirements. However, the requirement of large number of examples for training and the slow rate of convergence [17] for complex problems makes it unsuitable for many real world applications at this time.

In addition to these four major paradigms there are other sub paradigms such as learning by discovery [20] learning by experimentation [21], and learning by instruction [22], which are not studied extensively to derive useful applications. A more detailed treatment of the various machine learning paradigms is presented by Carbonell [2] and in the other papers [6, 9, 11, 15, 17] in the particular special issue of the journal. Carbonell concludes that connectionist paradigms are appropriate for learning in unstructured continuous domains with many training examples. Analytical paradigms at the other end are best suited in domains with rich structured knowledge even if only a few examples are available. Inductive and classifier systems bridge the gap between these two extremes.

3. CIVIL ENGINEERING APPLICATIONS

Literature on the application of machine learning in the civil engineering domain problem are very few. Rooney and Smith [22] discussed a feed back mechanism based model to two case studies covering the design of single span simply supported wide flange beams. Similarly many researchers have resorted to storing non-synthesized data from past experiences in a database for future reference. Such techniques are practically useless when the past experience is not much and becomes computationally inefficient when the number of stored examples increase.

Maher and Li [23] have demonstrated learning of default values, ranges for variables, relationship among numerical valued variables and patterns among nominal valued variables, using dependency network of earlier problem solving experience. However, the problem of when and how the decisions are made to perform the learning is not addressed in the paper.

Navinchandra et al. [24] have illustrated the role of analogues, heuristic rules, observed effects and engineering principles in problem solving through an example of a lever problem. The learning algorithm illustrated in the paper is conceptual and can not be extended to serious engineering application readily. Zhao et al. [25] used transformational analogy and similarity metric to retrieve solution to new problems from closely matching building examples in database. Murlidharan et al. [26] have used learning algorithms based on induction and analogical reasoning. These strategies create only a database that reduces the subsequent search space used to generate alternate configuration.

Arciszewski and Ziarko [27] have presented rough sets approach to inductive learning in civil engineering. The system extracts decision rules which can be used to acquire knowledge for problem solving to develop shallow model, to identify governing rules in a domain and to develop learning expert systems. Yeh et al. [28] have used the ID3 inductive learning algorithm to acquire diagnostic knowledge about the damage to PC piles while driving.

Adeli and Yeh [29] have demonstrated perceptron learning model for simple engineering design. This algorithm works for very simple tasks, which are trivial in engineering design, whereas this algorithm can not learn complex tasks since there are no hidden layers. Kamarthi et al. [30] have demonstrated a neural network learning system for vertical formwork selection. The paper discusses the merits and demerits of the neural network system when compared to rule based system and demonstrates that the difficulty of eliciting knowledge for rule based system can be overcome by the neural network learning system. Moselhi et al. [31] have illustrated neural network applications in the field of bidding for construction projects.

The examples of application of machine learning in the civil engineering domain clearly illustrate growing capability and complexity. Inductive learning methods are the most frequently used. Applications using neural network methods are being explored more recently. Applications in analytical methods and classifier systems are least explored, probably due to their computational complexity and application interface problems. It is also clear that applications so far discussed deal with only narrow domains of engineering problems.

4. MACHINE LEARNING IN ENGINEERING OF STEEL STRUCTURES

According to Simon [1] large knowledge based AI systems, particularly systems that can be expected to continue to grow and accumulate over a period of years of use, are fertile areas of application of machine learning. Engineering of steel structures is a large problem domain involving conceptual design, structural system planning, preliminary sizing, detailed analysis, design, document preparation and construction planning. The attributes in the domain represent the solution at various levels of a abstraction. Furthermore, the development of CAD system in the domain is incremental involving group effort. The system should model and accommodate the cooperative problem solving behaviour of domain experts working together. The machine learning in such an environment should be able to handle the varied requirements of the large domain. An integrated engineering system (IES) for the knowledge based engineering of steel structures has been already developed [32] under an ongoing project. The development and implementation of machine learning strategy in this system is currently under progress. The details of machine learning in this system are discussed in the following sections.

4.1 IES: Integrated Engineering System

IES uses an extended blackboard shell. Before discussing the machine learning implementation on this system, basic features of the systems are briefly reviewed [32]. Fig.1 shows the architecture of IES. The knowledge represent various functional activities of the engineering process, are compiled as production rules in independent knowledge sources. The knowledge sources generally do not interact directly but only through the global data referred to as blackboard. The blackboard has two panels namely solution blackboard and Control blackboard. The solution blackboard contains hierarchy of objects of the solution space with named links for inheritance. Objects and their attributes are represented as frames. Instances of the objects are stored in a relational database with links to blackboard objects. The control blackboard contains the status of the abstracted events of the solution process. Since the engineering process involves a large number of computations which is more efficiently carried out using algorithmic programs, C functions are used for such procedural programs. These function can be called from production rules in the knowledge sources. Generation of dependency network which is used in knowledge based backtracking, domain specific knowledge based control, opportunistic scheduling of knowledge sources are the other features of the system. More details about the system are presented by Sakthivel et al. [32].

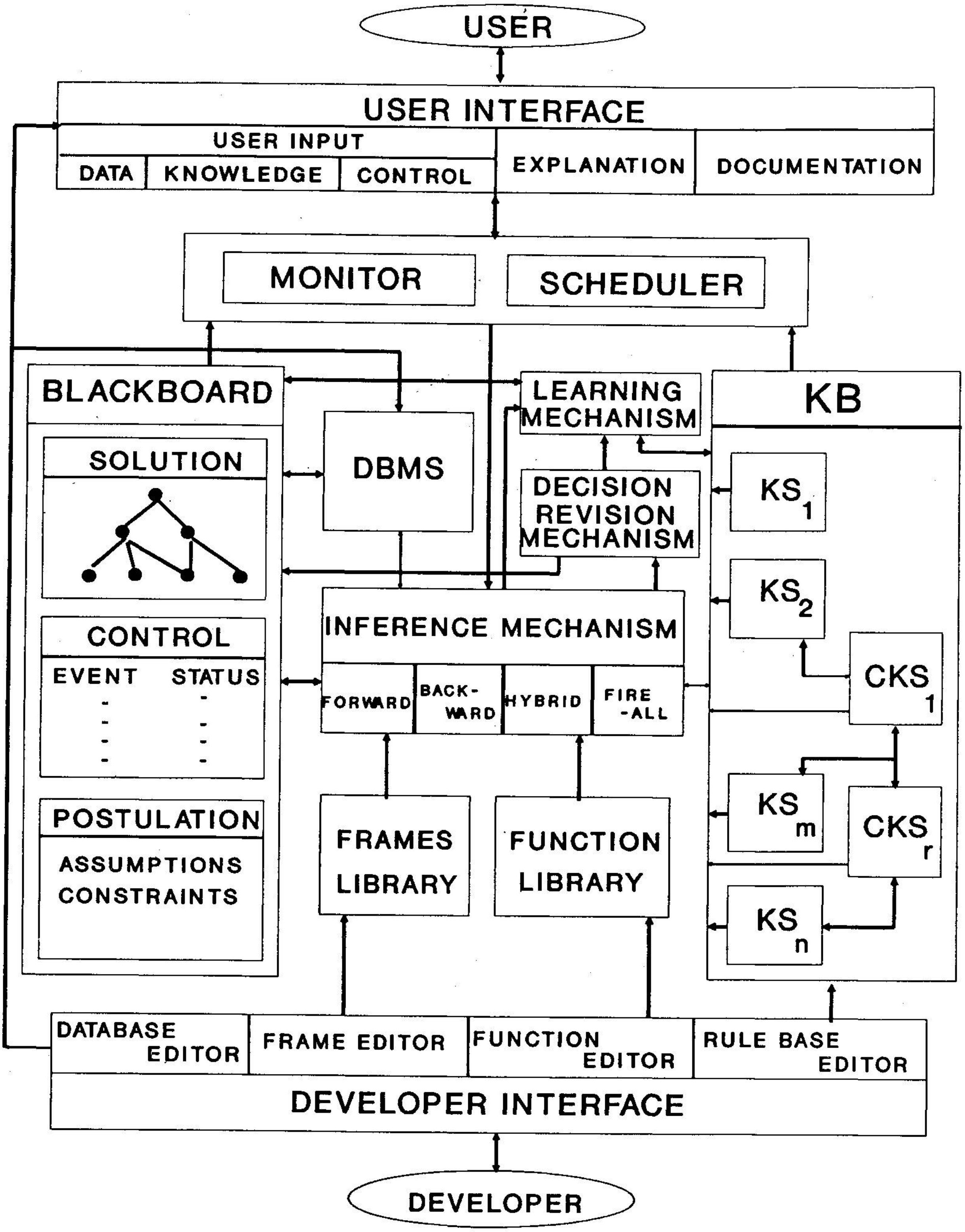

FIG.1. INTEGRATED ENGINEERING SYSTEM

4.2 Scope for Machine Learning

In large systems it is neither desirable nor feasible to consider machine learning as the backbone of the system. In engineering domain whenever problem solving steps and the engineering fundamentals that form the basis of the problem are well understood, it is efficient to represent such knowledge algorithmically in procedural programs. These are segments where governing knowledge is clearly defined or is easily acquired. However, the sequence of application of the knowledge during problem solving may be either not clear or has to be flexible. Knowledge based approach is more appropriate under such circumstances wherein the knowledge may be represented in production rules, frames, semantic networks, etc. Trying to acquire such knowledge for problem solving through machine learning is devious, unproductive and inefficient. However, scope exists for machine learning in engineering problem solving. Frequently knowledge or expertise is difficult to acquire and codify in which case machine learning from earlier problem solving experience can be of immense help. Besides, adaptation and modification of theory and practice is a continuing process in engineering problem solving. Engineering solution is affected by temporal, geographical and economic factors in non-obvious ways. Machine learning capabilities could synthesize such knowledge from past experience and help the system to adapt to changes in the theory and practice.

4.2.1 Engineering Tasks and Learning Strategies

In this section we discuss specific machine learning strategies that are being tried at various stages of engineering problem solving using IES. The preliminary specification of a structural engineering problem is brief open ended and ill-structured. Conceptual design based on this problem statement leads to an appropriate structural system, such as the type of bridge appropriate for a given specification being a cable stayed bridge or a truss bridge, etc. Decisions made at this stage have probably the greatest impact on the final economy of the engineering solution. However, the knowledge that drives the conceptual design and the application of the knowledge to arrive at appropriate decisions are not well understood. It is usually difficult to acquire the conceptual design knowledge. Analytic paradigms, such as case based learning or derivational analogy, are appropriate strategies for learning conceptual design. A few learning examples along with rich underlying domain theory support the learning task. Conceptual design is highly sensitive to temporal and geographical conditions. Hence a continuous learning system which could pursue multiple solution path and learn from each problem solving episode would be more robust. The frame based representation of objects, the dynamic instantiation of the objects in the solution space and events in the control blackboard, as well as knowledge based control strategy are the features of the IES system, which readily support the analytic learning strategy.

Having decided on the structural system, planning and configuring the structural subsystems is the next step in the engineering process, which offers opportunities for the machine learning process. Maher and Li [23] have demonstrated conceptually, a learning system for configuring cable stayed bridges based on the dependency network of the design experience. Inductive paradigms such as conceptual clustering using a sequence of known examples and counter examples from previous problem solving sessions support this process. This paradigm is being tried for the structural configuring activity in IES.

Let us consider the task of learning the configuration generation of a cable stayed bridge from a number of cases already engineered. The attributes that define the configuration of a cable stayed bridge may be subdivided into problem specification attributes and configuration attributes to be generated by the system. The specification attributes are the total length of bridge, number of lanes of traffic, geotechnical details of the site, navigational requirements under the bridge, wind and earthquake load at the site, approach alignment, and aesthetic requirements. The configuration attributes are the number of cable stayed spans, maximum span length, side span length, drop span, tower type, tower height, number of cable planes, inclination of cable planes, number of cables per span, cable arrangement, girder type, girder depth, and foundation type. Cases of cable stayed bridges are available in the literature [35] which could be used as learning examples in induction. Inductive paradigms based on concept acquisition [6] require tutoring and would not serve the requirements. Conceptional clustering CLUSTER/2 [36] and other similar algorithms can generate only a hierarchical organisation of objects classified by conjunctive statements. The learning process in the configuration generation should be able to represent many to many relationship between objects derived using operators expressing other logical implication in addition to conjunction. An induction algorithm which can create a network structure between attributes of the domain. This would involve creation and use of fuzzy definition of attribute values.

Decisions regarding trial shapes and sizes for members are made at the stage of preliminary sizing. Past experience plays a major role at this stage. Maher and Li [23] and Adeli and Yeh [24] have demonstrated machine learning in this domain using induction and perceptron, respectively. Neural network with hidden layers could learn from earlier design experience and thus enhance preliminary design capability.

Detailed design is the iterative process of checking the adequacy of trial sections to meet all the constraints of the design. This falls under the category of routine design. The detailed design has to be repeated for many member in the structural system such as tension and compression members of the truss bridge as well as their connections. Knowledge chunking algorithm helps in speeding up this process [33].

Time and cost overrun in large projects are frequently due to the difficulty in planning and managing such construction projects. Technical, social and environmental uncertainties influence the construction process. Construction planners and managers learn to tackle these activities under uncertainties, based on their past experience on similar projects. Cause effect relationship in these activities is not well documented. Neural network system can be trained using past cases to learn the implicit knowledge associated with the process. The trained neural network serves as the transfer function relating inputs and outputs of the construction planning process. The self organisation, generalisation, fault tolerance, and massively parallel processing properties of the neural network systems are useful in this activity.

The process of solving any major engineering problems is an open ended problem. Many agents cooperating opportunistically, and interacting in a non-deterministic and non-trivial way contribute to the solution in an incremental but non-monotonic fashion. Computing systems such as DICE [34], attempt to facilitate such a cooperative problem solving process in real time. IES based on opportunistic knowledge scheduling, models such a problem solving strategy. In IES without learning capabilities, the choice of one rule from among many competing rules or one knowledge source from among the competing knowledge sources is predetermined by the priorities set in advance, based on the experience of the developer. In IES with learning capabilities such priorities can be continually updated based on the past problem solving experience. The process and not the product of the past experience is used in learning. Induction paradigms provide algorithms for learning problem solving process.

IES also has a rich user interaction facility. Control is given to the user whenever a new input is required or a new knowledge source is to be scheduled in addition to pauses at pre-defined points depending upon the domain requirements. During such interruptions the user can review the solution and make modifications to any value already inferred or change the event to be pursued which may be different from that dictated by control knowledge. Such user inputs serve as a rich source for learning the problem solving process. An abstraction of the entire problem solving trace is stored in IES as a dependency network. The dependency network also serves as a source for learning the problem solving process. An induction learning algorithm could be used to achieve this learning. IES handles the non-monotonic problem solving process in engineering, using the dependency network and the consistency maintenance mechanism. Whenever a design failure or a constraint violation occurs, the knowledge based backtracking mechanism takes over and restarts from an earlier state after appropriate modifications to the solution state and dependency network. The knowledge for the backtracking may be available in the knowledge base, if the episode has been already envisaged. Otherwise the advice is obtained from the user. Such backtracking knowledge with accompanying explanation is to be used to minimize or eliminate unnecessary problem solving cycles in the subsequent sessions in the IES, using an explanation based learning algorithm.

5. SUMMARY AND CONCLUSIONS

It is seen that no single strategy could effectively serve the machine learning requirements of large applications. IES requires implementation of different learning strategies and the engineering application developer can make a choice depending upon the domain requirements. The opportunistic knowledge scheduling and maintenance of dependency network in the IES system based on extended blackboard architecture are features which aid the implementation of the learning strategies. The learning strategies as discussed are being currently implemented and tested in the IES system. For brevity, implementation details are not presented in this paper.

6. ACKNOWLEDGEMENT

The work reported in this paper is supported by the sponsored research project on "Expert System Based Computer Aided Engineering of Steel Structures", supported by Department of Science & Technology.

REFERENCES

1. SIMON H.A., Why Should Machines Learn, in Michalski, R.S. Carbonell J.G. and Mitchell T.M. (Eds), Machine Learning, Tioga, Palo Alto CA, 1983, pp. 25-37.
2. CARBONELL J.G., Introduction: Paradigms for Machine Learning, Artificial Intelligence, Vol. 40, No. 1-3, 1989, pp. 1-9.
3. DIETTERICH T.G. and MICHALSKI R.S., Inductive Learning of Structural Descriptions, Artificial Intelligence, Vol. 16, No. 3, 1981, pp. 257-294.
4. HUNT E.B., Concept Learning:An Information Processing Problem, Wiley, New York, 1962.
5. QUINLAN J.R., Induction of decision trees, Machine Learning, Vol. 1, 1986, pp. 81-106.
6. GENNARI J.H., LANGLEY P. and FISCHER D., Model of Incremental Concept Formation, Artificial Intelligence, Vol. 40, No .1-3, 1989, pp. 11-61.
7. HALL R.J., Computational approaches to analogical reasoning:A comparative analysis, Artificial Intelligence, Vol. 39, No. 1, 1989, pp. 39-120.
8. WINSTON P.H., Learning and Reasoning by Analogy, Commn. ACM, Vol. 23, No. 12, 1980, pp. 689-703.
9. MOSTOW J., Design by Derivational Analogy:Issues in the Automated Replay of Design Plans, Artificial Intelligence, Vol. 40, No. 13, 1989, pp. 119-184.
10. KOLODNER J., SIMPSON R. and SYCARA K., A process model of case- based reasoning on problem solving, in Joshi A. (Ed)., Proc. of IJCAI-85, Los Angeles, CA, 1985, pp. 284-290.
11. SCHANK R.C. and LEAKE D.B., Creativity and Learning in a case- based Explainer, Artificial Intelligence, Vol. 40, No. 1-3, 1989, pp. 353-385.
12. DEJONG J. and MOONEY A., Explanation Based Learning - An alternative view, Machine Learning, Vol. 1, 1986, pp. 145-176.
13. MINTON S., CARBONELL J.G., KNOBLOCK C.A., KUOKKA D.R., ETZIONI O., AND GIL Y., Explanation Based Learning:A Problem solving perspective, Artificial Intelligence, Vol. 40, No. 1-3, 1989, pp. 63-118.
14. CARBONELL J.G., Learning by Analogy:Formulating and generalizinG plans from past experience in Michalski R.S., Carbonell J.G. and Mitchell T.M. (Eds.), Machine Learning , Tioga, Palo Alto, CA, 1983, pp. 137-161.
15. BOOKER L.B., GOLDBERG D.E. and HOLLAND J.H., Classifier systems and Genetic Algorithms, Artificial Intelligence, Vol. 40, No. 1-3, 1989, pp. 235-282.
16. HOLLAND J.H., Genetic algorithms and the optimal allocation of trials, SIAM Jl. Comput., Vol. 2, 1973, pp. 88-105.
17. HINTON G.E., Connectionist Learning Procedures. Artificial Intelligence Vol. 40, No. 1-3, 1989, pp. 185-234.
18. AMARI S.I., A theory of adaptive pattern classifiers, IEEE Trans. Electron. comput., Vol. 16, 1967, pp. 299-307.
19. LENAT D.B., The role of heuristics in learning by discovery:three case studies in Michalski R.S., Carbonell J.G. and Mitchell T.M. (Eds.), Machine Learning , Tioga, Palo Alto, CA, 1983, pp. 243-306.

20. MITCHELL T.M., UTGOFF P.E. and BANERJI R., Learning by experimentation:Acquiring and refining problem solving heuristics, in Michalski R.S., Carbonell J.G. and Mitchell T.M. (Eds.), Machine Learning, Tioga, Palo Alto, CA, 1983, pp. 163-190.
21. MOSTOW D.J., Machine transformation of advice into a heuristic search procedure, in Michalski R.S., Carbonell J.G. and Mitchell T.M. (Eds.), Machine Learning, Tioga, Palo Alto, CA, 1983, pp. 368-403.
22. ROONEY M.F. and SMITH S.E., Artificial intelligence in engineering design, Computers & Structures, Vol. 16, No. 1-4, 1983, pp. 279-288.
23. MAHER M.L. and LI H., Automatically Learning preliminary design knowledge from design examples, Microcomputers in civil engg., Vol. 7, No. 1, 1992, pp. 73-80.
24. NAVINCHANDRA D., SRIRAM D. and KEDAL-CABELLI S.T., Analogy based engineering problem solving:An overview, in KBES in engineering:Tools and techniques, 1987, pp. 273-285.
25. ZHAO F. and MAHER M.L., Using analogical reasoning to design buildings, Engg. with computers, Vol. 2, 1988, pp. 107-119.
26. MURALIDHARAN T.L., ARAVIND H.B., SURYAKUMAR G.V., AND RAMAN N.V., Expert tower analysis and design systems:Part II. Search strategies and learning, Jl. of comput. in civil engg., Vol. 5, No. 2, 1991, pp. 193-210.
27. ARCISZEWSKI T. and ZIARKO W., Inductive learning in civil engineering:Rough sets approach, Microcomputers in civil Engg, Vol. 5, No. 1, 1990, pp. 19-28.
28. YEH, Y.C., Kuo Y.H. and Hsu D.S., Building KBES for diagnosing PC Pile with inductive learning, Jl. of comput. in civil engg., Vol. 6, No. 2, 1992, pp. 200-219.
29. ADELI H. and YEH C., Perceptron learning in engineering design, Microcomputers in civil engg., Vol. 4, No. 4, 1989, pp. 247-256.
30. KAMARTHI S.V., SANVIDO V.E. and KUMARA S.B.T., Neuroform - neural network system for vertical formwork selection, Jl. of comput. in civil Engg., Vol. 6, No. 2, 1992, pp. 178-199.
31. MOSELHI O., HEGAZY T. and FAZIO P., Neural networks as tools in construction, Jl. of constrn. engg. & mgt., Vol. 117, No. 4, 1991, pp. 606-625.
32. SAKTHIVEL T.S. and KALYANARAMAN V., A KBES for integrated engineering (Accepted for publication in Engineering with Computers)
33. REICH, Y. and FENVES S.J., Floor-system design in SOAR:A case study of learning to learn, Technical report EDRC-12-26-88, Engg. Design Research Center, Carnegie Mellon University, Pittsburgh, PA, 1988.
34. AHMED S.M., SRIRAM D., AND LOGCHER R., Transaction Management issues in collaboration engineering, Jl. of comput. in civil engg., Vol. 6, No. 1, 1992, pp. 85-105.
35. PODOLNY W. and SCALZI J.B., Construction and design of cable- stayed bridges, Wiley Interscience, U.S.A, 1986.
36. MICHALSKI R.S. and STEPP R.E., Learning from observation:Conceptual clustering, in Michalski R.S., Carbonell J.G. and Mitchell T.M. (Eds.), Machine Learning , Tioga, Palo Alto, CA, 1983, pp. 331-363.