A KNOWLEDGE-BASED SYSTEM FOR CASH MANAGEMENT:

WITH IMPLICATIONS FOR STRUCTURING DSS AND WITH EXTENSIONS FOR SYSTEM LEARNING AND KNOWLEDGE ACQUISITION

Daniel E. O'Leary and Todd Eis

ABSTRACT

The purpose of this paper is threefold. First, it discusses a knowledge-based system designed to aid the cash management process. Cash diagnosis system (CDS) is a heuristic-based system that uses rules based on financial ratios for industry comparison, firm-based historical and budget comparisons, loan covenant restriction analysis, bankruptcy analysis, and ongoing assessment of the possibility of the firm obtaining a loan, in order to diagnose cash problems.

CDS is a diagnostic system that can be used to monitor, in an ongoing manner, the financial information in a firm. It also can be used to ascertain the impact of

Advances in Working Capital Management, Vol. 2, pages 197–209.

ISBN: 0-89232-996-3

alternative strategies that have differing, yet predictable financial statement consequences.

Second, this paper suggests that the generic purpose of an expert system can be used to guide the integration of expert systems into decision support for cash management purposes. This allows us to add another dimension to a framework of decision support for cash management.

Third, this paper also discusses some important extensions to CDS that are applicable to other financial systems. In particular, it examines how recent advances in computer learning theory can be adapted to help develop knowledge bases for these systems and to help these systems adapt to changing conditions.

I. INTRODUCTION

One of the most important problems facing financial management is cash management. A critical part of cash management is concerned with diagnosing potential cash problems and planning solutions to managing the firm's cash flows.

The purpose of this paper is threefold. The first purpose is to discuss a knowledge-based system for analyzing potential cash problems. The system, cash diagnosis system (CDS) uses rules based on financial ratios for industry comparison, firm-based historical and budget comparisons, loan covenant restriction analysis, bankruptcy analysis, and ongoing diagnosis of potential possibility of borrowing. CDS provides a "proof of concept" that a system can be built to accomplish these important interrelated judgmental purposes, for cash management, using heuristic ratio-based knowledge.

CDS can be used to monitor the financial operations of the firm in an ongoing and unobtrusive manner in support of financial management. Alternatively, CDS can be used to ascertain the impact of plans that can be characterized by financial consequences.

The second purpose is based on categorizing CDS as a system designed to diagnose. This paper suggests that the generic purpose of the system may be of use in the development of a decision support system (DSS) for cash management. As a result, this paper discusses an extension to the framework of Srinivasan and Kim [19] to include the generic purpose of the system.

The third purpose is to analyze how systems like CDS can be extended to include learning or adapting to new situations. Symbolic-concept acquisition and genetic learning algorithms are investigated in an effort to bring those properties to CDS.

The paper proceeds as follows. Section II briefly reviews certain concepts of DSS, knowledge-based systems, and artificial intelligence (AI). Section III discusses some of the previous research in decision support of cash management problems and an extension to the framework of Srinivasan and Kim [19]. Section

IV summarizes the overall structure of CDS. Then Sections V and VI discuss portions of that structure, in particular, the database and the knowledge base. Section VII analyzes the integration of learning and adaptation into CDS. The last section summarizes the paper.

II. DSS, KNOWLEDGE-BASED SYSTEMS, AND AI

DSS are computer-based systems aimed at supporting management decision-making (Keen & Scott Morton [13]). Typically, DSS make a portfolio of models and interfaces to those models, e.g., mathematical programming algorithms or graphics, available to the user to aid decision-making. In some cases, DSS include AI applications in their portfolio of models. For example, Keen and Scott Morton [13] describe the well-known AI-based system, EMYCIN (Buchanan and Shortliffe [5]), as a DSS. In addition, most AI-based systems are used to support, rather than replace decision-makers.

AI is that part of computer science aimed at developing computer programs that perform tasks requiring "intelligence," that is, "systems that exhibit the characteristics we associate with intelligence in human behavior—such as understanding language, learning, reasoning, solving problems and so on" (Barr & Feigenbaum [2, p. 3]). AI includes a number of subfields (Barr & Feigenbaum [2, p. 9]), among them language robotics, learning and reasoning, pattern recognition, expert systems. Expert systems (ES) are computer programs designed to capture aspects of human expertise. Because few ES actually capture "true expertise" some researchers prefer to refer to systems that use aspects of expertise as knowledge-based systems. However, this paper uses those terms interchangeably.

Structurally, an ES usually consists of four major components: a database, a knowledge base, an inference engine, and a user interface. The database contains the data used by the ES. The data may be built into the system, they may be elicited from the user by the system, or they may be in a separate database. These generally are the same data that the human expert would use in the solution of the problem. However, the ES may use different data than the human expert. On the one hand the human expert may use additional uncertain and fuzzy data not easily built into the ES. On the other hand, the system may employ unequivocal data that a human could not process in a timely manner.

The knowledge base provides the set of knowledge that the expert uses to process the data (thus the name "knowledge-based" system). Typically, this is the domain-specific knowledge that the expert would use to solve the problem. Knowledge can be represented a number of ways. One of the most frequently used methods is the rule-based approach. Rule-based knowledge representation generally takes the form, "if (condition) then (consequence/goal)." The rules may or may not include a numeric level of confidence or probability of occurrence.

The inference engine provides the basis to use the knowledge base to process the database. In a rule-based system, the inference engine normally uses either a forward- or backward-chaining approach. Forward chaining reasons toward a goal. Backward chaining reasons backward from the goal to determine if or how the goal can be accomplished.

The user interface defines the manner in which the system elicits information and explains its conclusions. Generally, this interface is friendly, particularly when the system must help the user determine the relevant input information. Further, the interface may include output to the user that explains the reasons for its answers.

A. ES Shells and AI Languages

Usually, ES are developed using either an AI language or an ES shell. An AI language is a computer language aimed at processing symbolic information. Two of the primary AI languages are Prolog and Lisp. Prolog is the language chosen by the Japanese for their fifth-generation computer. Lisp has been used extensively in AI in the United States.

An ES shell is software designed to aid in the development of ES. Typically an ES shell makes the storage of knowledge easier and prespecifies the inference engine. ES shells can shorten the ES development cycle. The ES shells also may allow the user to access existing databases (such as dBase III), spreadsheets (such as Lotus), and AI languages (such as Lisp). There are a number of ES shells, for example, M.1, Personal Consultant Plus, and EXSYS.

B. Uses of ES

Hayes-Roth et al. [9, p. 14] outline ten different generic categories of ES applications, based on decision-making and expertise:

- Interpretation: inferring situation descriptions from sensor data
- Prediction: inferring likely consequences of given situations
- Diagnosis: inferring system malfunctions from observables
- Design: configuring objects under constraints
- Planning: designing actions
- Monitoring: comparing observations to plan vulnerabilities
- Debugging: prescribing remedies for malfunctions
- Repair: executing a plan to administer a prescribed remedy
- Instruction: diagnosing, debugging, and repairing student behavior
- Control: interpreting, predicting, repairing, and monitoring system behaviors

As with many of the more successful ES, the system described in this paper is diagnostic in nature. For example, it analyzes a selected set of cash and working capital ratios to determine if the cash system is "malfunctioning."

III. DECISION SUPPORT FOR CASH MANAGEMENT

Recently, Srinivasan and Kim [19] surveyed the cash management literature and provided a framework for decision support for cash management. As they note, the academic literature primarily has provided sophisticated tools to solve isolated and well-structured problems that ignore the broader constraints that cash managers face. They develop a framework that ultimately structured cash management by the type of decision: operational or infrastructural. Operational decisions are made on a daily basis as new information becomes available, whereas infrastructural decisions relate to the establishment of an organizational structure through which the company's resources are channeled. These decisions are made less frequently than operational decisions. Srinivasan and Kim argue that by characterizing the decisions in this manner more integrated approaches to decision support of cash management can be attained.

The ES developed in this paper captures some of that same spirit of integrating applications. It contains information on a number of problems that are normally treated independently of each other, e.g., bankruptcy analysis, loan covenant analysis, and industry comparison.

Although the ES discussed here provides operational support, ES may ultimately be used to support infrastructure decisions. For example, they may be used to analyze infrastructure notions of lockbox design, deposit bank selection, disbursement site selection, and other aspects of cash management problems, by capturing the expert judgment of those who make such decisions.

A. Generic Uses of ES

If we view a DSS as a portfolio of tools to support decision-making, then the notions of the generic uses of ES can provide further understanding of the characteristics of that portfolio of tools. In the two types of decision-making problems summarized in Srinivasan and Kim [19] it appears that decision support tool needs can vary as a function of generic uses and generic problem type. For example, in *operational-prediction* decisions, the planning tools would be short run oriented. The models would probably be based on time-series or other statistical models. However, in *infrastructural-prediction* problems, longer–time range models would be used. Accordingly, they may employ results from, e.g., a Delphi study or similar model. Similarly, many of the *operational-planning* models likely would be one-period models. On the other hand, in *infrastructural-planning* decision-making problems there may be more of a multiperiod nature to the problems. Other types of DSS tools could be used to accomplish different cash management purposes. In a similar manner, the cash manager also must interpret, plan, predict, design, monitor, repair, and control. Other examples can be developed for each of those generic purposes.

IV. STRUCTURE OF CDS

CDS was developed using the expert system shell EXSYS. Thus, the inference engine and the user interface are prespecified.

The database of CDS is contained in a Lotus spreadsheet. The firm's financial information is summarized in traditional accounting report formats of income statement and balance sheet. Then the capabilities of the spreadsheet are used to generate the necessary ratios that are required, as a variable database, for the knowledge-based system, CDS.

The knowledge base was developed using the expert system shell EXSYS. The knowledge base consists of a number of if–then rules summarizing the ratio analysis relationships. Data to be analyzed are gathered from the spreadsheet by the ES shell.

The system analyzes the information and provides the user with diagnostic information about the current and expected state of cash resources. It also provides the user with information on other issues of importance to the firm, e.g., the possibility of getting a loan based on the current state of the firm's liquidity.

New rules can be added to the knowledge base relatively easily. Thus, additional knowledge, e.g., new ratio information, can be placed in the knowledge base as long as the necessary financial information relating to that knowledge is in the database. Otherwise, similar to a human expert, the system would be unable to use the knowledge.

V. DATABASE FOR CDS

The basic database for CDS consists of the financial information of the company, the industry standard of comparison, and the financial budget or estimate of future behavior. CDS uses financial information available from the income statement and balance sheet. The system uses that information to develop the ratios that are used in the system. Since the system contains substantial amounts of financial data, other financial ratios can be added to the system.

A. Industry Standards for Ratios

Chudson [6] found that liquidity and turnover ratios were affected significantly by the industry grouping. Horrigan [12] corroborated those findings and found that size of firm also affected the liquidity ratios. Further, the accounting methods have been found to affect financial ratios (e.g., Holdren [10]). Thus, the choice of the standards for comparisons in different industries should consider not just the industry, but also the size of the firm and accounting methods. In addition, the choice of whether average performance or better performance should be used as the industry standard will affect the relative rating of the firm. Since the industry standard is a part of the database of CDS, the resolution of such issues is ultimately up to the specific user.

B. Firm Budgets and Forecasts

Firm budgets or forecasts of firm performance can provide some insight into the future performance of the firm and the firm's ratios. However, there are many approaches to developing such estimates of future behavior. Although these estimates are a part of the database of CDS, such issues as the development of these estimates are in the hands of the user.

VI. KNOWLEDGE BASE FOR CDS

A. Scope

CDS is generic in nature. Each of the ratios is a generally accepted measure and does not take advantage of the specific nature of the industry. For example, in the railroad industry, many specific measures relate to miles of track. Although CDS does not utilize such measures, the system could be expanded to include industry-specific ratios or other measures.

B. Nature and Importance of Ratio-Based Knowledge

CDS employs a number of heuristics based on ratio knowledge. The importance of financial ratios in the assessment of firm behavior is well-established. This is verified by the large number of financial service organizations that provide financial ratio information to their subscribers, by the large number of books that discuss ratios, and by the large number of research papers that discuss, e.g., the predictive power of ratios (Beaver [3]).

Although ratios provide quantitative measures, this type of knowledge is not entirely cut and dry. The use of ratios in the analysis of financial-based situations can require substantial judgment:

> Ratios are tools of analysis which, in most cases, provide the analyst with clues and symptoms of underlying conditions. Ratios properly interpreted also can point the way to areas requiring further investigation and inquiry. The analysis of a ratio can disclose relationships as well as bases of comparison which reveal conditions and trends that cannot be detected by an inspection of the individual components of the ratio. (Bernstein [4, p. 71])

C. Ratio Interpretation

CDS uses financial ratios to guide the analysis of the adequacy of cash balances. This is consistent with other users of financial ratios (Cohen et al. [7], Lev [14], Bernstein [4], and others).

Unfortunately, as noted in Lev [14], the literature on ratio analysis is "rather vague regarding the interpretation of ratios." As a result, this paper makes a few assumptions about ratio analysis. First, it is assumed that the firm wishes to have financial ratios similar to those of other firms in the same industry. This is not an unusual assumption, since some researchers have noted that firms appear to try to move toward industry standards and that generally lenders seem to expect such behavior. Second, the effect of ratios is measured, not on an absolute basis, but as a percentage of the industry ratio. This is consistent with the whole notion of a ratio normalizing absolute differences. Third, it is assumed that within some range of the industry ratio, both plus and minus, there is no need for the concern of cash managers. Fourth, whenever, the firm's ratio is either too large or too small then the system diagnoses the existence of a problem. This analysis is done while taking into account the "liquidity preferences" of the user. (This is discussed in more detail below.) Fifth, this paper assumes that both univariate and multivariate analysis (in the form of indices) of the ratios are helpful, although most of the discussion is concerned with univariate analysis.

D. Ratio Knowledge in CDS

As noted by Bernstein, ratios can be used in a number of different ways:

> Ratios, like most other relationships in financial analysis, are not significant in themselves and can be interpreted only by comparison with (1) past ratios of the same enterprise, or (2) some predetermined standard, or (3) ratios of other companies in the industry. The range of the ratio over time also is significant as is the trend of a given ratio over time. [4, p. 73]

Since CDS uses only two years of data, the range of the ratio is subsumed by comparison to the previous year. However, CDS uses ratios in these prescribed ways and in other ways. First, CDS uses a univariate analysis of individual ratios, based on forecasted data, to determine if the ratios are within a tolerance range of the industry standard. CDS includes a sensitivity measure that defines a "tolerance interval" in which managers are indifferent. For example, if the sensitivity is equal to 0.10, the forecasted ratio was 1.5, and the (target) industry ratio was 2.0, then CDS would bring this to the attention of management, because 1.5 is outside of 0.10 tolerance range (2.0 ± 0.2).

CDS does not assume symmetric utility functions. For example, generally, most managers would indicate that a current ratio more than 10% above the industry current ratio, say 15%, would be satisfactory, while fewer managers would indicate that an equal 15% below the industry standard was satisfactory. CDS allows the user to build this lack of symmetry into the system. This is done by use of liquidity preference measure (LPM), which is multiplied times the tolerance measure. Thus, LPM = 1 would leave the tolerance measure unchanged, while LPM = 1.5 would change the tolerance range, on the high side of the ratio, from 0.10 to 0.15.

Second, CDS does not just ascertain if the ratios are within the tolerance range. It also analyzes the movement of the ratios. This is done by relating the ratios derived from forecasted data to historical data. Three categories are used to characterize the movement of the ratios, although this can be generalized to an arbitrary number of categories. It is assumed that managers wish to manage their ratios toward industry standards. Thus, there can be no movement, there can be movement toward those standards ("good movement"), and there can be movement away from the standards ("bad movement").

User-defined sensitivity and liquidity preference also is used in this segment of CDS. The industry standard ratio and the historical ratio are compared to the forecasted ratio to determine the movement. If the forecasted ratio is within the defined sensitivity and liquidity preference of the historical ratio, then the system records that there was no material movement. If the forecasted ratio is outside the sensitivity and liquidity preference level of the historical ratio then there is either good or bad movement.

For example, assume sensitivity is defined to be 0.10, LPM = 1.5, the forecasted ratio is 2.0, the historical ratio is 3.0, and the industry ratio is 3.5. Thus, the sensitivity range is between 2.7 and 3.45 ($3 - 3 \cdot 0.1$ and $3 + 3 \cdot 0.15$) so that the forecasted ratio is outside the tolerance range, based on the historical ratio. In addition, it is moving in a direction away from the industry standard; thus, there is bad movement.

Third, CDS monitors the behavior necessary to determine if the firm is satisfying its loan covenant agreements. Restrictive covenants are often in ratio format. CDS computes and then determines if the loan covenant ratios are being satisfied.

Fourth, CDS monitors the possibility of bankruptcy, given the forecasted financial information. Bankruptcy, or even the possibility of bankruptcy, can cause trauma for a firm's managers, investors, suppliers, customers, and the community. Thus, it is beneficial to be able to predict bankruptcy so that steps can be taken to avoid it or at least mitigate its impact.

There is evidence that the behavior of cash and other types of working capital is indicative of the possibility of bankruptcy. Bankruptcy can be forecasted in a number of ways. There are both univariate and multivariate models. For example, Beaver [3] found that selected financial ratios, some based on cash and current assets, were useful in bankruptcy prediction. There also are a number of multivariate methods that can be used (e.g., Ohlson [15]). Because much of CDS is univariate in nature, a multivariate approach was chosen. Further, because of ease of implementation, Altman's [1] discriminant analysis approach was chosen. Using that approach, CDS estimates the possibility of bankruptcy and informs the user if it appears that the firm is facing bankruptcy.

Fifth, CDS includes knowledge about the possible assessment of the firm for a business loan. Researchers such as Cohen et al. [7], have investigated the decision-making procedures associated with making a business loan. They found that process can be characterized by a set of judgments based on financial information of the firm. Part of that set of judgments is based on liquidity. In particular, Cohen et al. [7, p. 242] found that in the analysis of loan applications, two of the primary questions were "Does the firm have enough current assets?" and "Are the firm's assets sufficiently liquid?" In order to perform that analysis, they established decisions based on financial information, including the following:

- net working capital
- percentile value of the current ratio
- current assets/current liabilities
- cash/total current liabilities
- inventory/current assets

CDS also includes analysis of this type to help monitor, in an ongoing manner, the capability of the firm to obtain a loan.

Sixth, initially CDS was to contain knowledge about the possibility of takeover due to the state of the current assets or cash. However, since there is no evidence that short-term assets are predictive of corporate takeover attempts (e.g., Simkowitz and Monroe [18]), the possibility of takeover is not monitored by CDS.

VII. LEARNING BY CDS

Unfortunately as currently configured, CDS does not employ multiple-ratio pattern recognition. For example, it does not analyze patterns of the current ratio and the quick ratio simultaneously, without aggregating multiple ratios into an index such as that generated from multiple-discriminant analysis. In addition, CDS does not learn from its past experience. It makes the same evaluations, given the same set of inputs, every time.

However, this need not always be the case. Recently, adaptive learning algorithms have been developed that permit the determination of such multivariate patterns. Such algorithms can be used to provide knowledge for systems such as CDS that employ financial data.

A. Symbolic-Concept Acquisition

Symbolic-concept acquisition refers to the establishment of patterns of conditions (possibly binary in nature) that lead to consequences. For example, in CDS there are a number of financial ratios that can function as conditions, when those ratios are compared to the selected industry standards. For example, a current ratio greater than 2.0, a quick ratio greater than 1.8, etc., form the basis of a given pattern of such binary conditions. Then the entire set of those conditions lead to a consequence, e.g., "take no additional action to increase cash balance," as noted by cash managers.

Quinlan [16,17] developed a classification system, ID3, that can be used to develop such if–then rules. Given a set of conditions and consequences as above, ID3 evaluates the ability of each condition to discriminate between the consequences. A decision tree is then built by eliciting the best single discriminating condition, the next best discriminating condition, etc., until the best set of discriminating conditions is found. "Best" refers to the development of a partitioning of the data so that the fewest total misclassifications is made. That set can then be implemented as a rule.

There can be limitations of this process. A "one condition at a time" approach may lead to a locally optimal rule that is not globally optimal. Further, with the use of total misclassifications as the objective function, there is an implicit assumption that type I and type II errors are equally as costly. In addition, "noise" in the data can camouflage a relationship or produce a nonexistent relationship. Thus, in some cases the symbolic-concept acquisition approach may yield poor resuits.

B. Genetic Algorithms

Genetic algorithms (Holland [11]) employ a concept of adaptive efficiency in the context of a probabilistic search method. They are called genetic algorithms because they employ methods analogous to genetic transfer from one organism to its offspring. They assume that, as in nature, the organisms that are best suited to the environment flourish and produce offspring with similar genetic traits.

Genetic search algorithms can be employed by representing the conditions of an if–then rule as a string of binary conditions. For example, a binary representation of the condition (current ratio > 2) would be 1 if it was true and 0 if it was not true. In order to find the best string, genetic search algorithms create a population of rules randomly and then evaluate the quality of the rules using a performance measure (discussed below). Rules that perform well survive and are used to create other rules, through breeding (discussed below). This process continues until either a preestablished number of generations have elapsed or some performance level is achieved. Since the binary conditions are used, other if–then rules can be created for the knowledge base.

"Breeding" provides the development of potential new rules for the system. There are a number of types of breeding operators. One is crossover, whereby given two rules with an equal number of conditions (say, m) two new rules are formed. This is done by using conditions $1, \ldots, i$ from one of the parents and conditions $i + 1, \ldots, m$ from the other parent. Another approach that also develops two new rules is to take a random selection of k ($<m$) from one of the parents and $m - k$ from the other, to form one rule, while the remaining conditions would be assigned to the other new rule. A number of such strategies can be used.

Performance rules are a critical part of the genetic algorithms. (A number of alternative performance rules are studied in Greene et al. [8].) Given a set of conditions, there must be a means by which to evaluate the quality of that set of conditions, i.e., the ability of that set of conditions to be able to determine that action should be taken to improve the status of the short-term assets. When dealing with financial data, the performance rule can be an index (e.g., based on Altman's [1] discriminant analysis). It could be based on the behavior of some single ratio that is not in the condition set, or human financial analysts' ranking of performance.

Information can be provided for the genetic search algorithms using actual financial data encountered over time by the system. For example, information relating to the entire set of conditions can be recorded, in conjunction with the judgment of the financial analyst regarding the status of the short-term assets. The judgment may be recorded unobtrusively by the system based on the actions that the analyst takes regarding the short-term assets.

Using an approach like this can allow systems such as CDS to employ patterns of ratios in its analysis. In addition, this approach could allow systems such as CDS to learn and adapt to changing situations.

VIII. SUMMARY

This paper has discussed a prototype knowledge-based system for diagnosing cash management problems. The system, CDS, can be used either to monitor ongoing operations or to test the implications of alternative strategies that have different and discernible financial statement consequences.

The system was developed using the ES shell EXSYS to capture the knowledge. The database was developed from financial statements and was maintained in Lotus.

The system employs multiattributes. The system monitors the relationship of ratios to industry norms, monitors the trends of critical ratios, monitors the firms loan covenant restrictions, analyzes the possibility of bankruptcy, and provides an ongoing analysis of the firm's position in obtaining a loan.

The current version of CDS uses univariate financial ratios and multivariate-based indices (developed from, e.g., discriminant analysis) to analyze the cash and working capital portion of the financial statements. This paper also presented a means to incorporate more complex patterns of financial ratios. Symbolic-concept acquisition and genetic algorithms also were extended for use in financial systems such as CDS.

The diagnostic nature of the system also suggests including the generic nature of the ES as an additional dimension in a decision-based framework for DSS in cash management.

REFERENCES

1. E. Altman. "Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy," *Journal of Finance*, September 1968, pp. 589–609.
2. A. Barr and E. Feigenbaum. *The Handbook of Artificial Intelligence*. Stanford, CA: Heuristech Press, and Los Altos, CA: William Kaufmann, 1981.
3. W. Beaver. "Financial Ratios as Predictors of Failure," *Empirical Research in Accounting: Selected Studies, 1966*, Supplement to the *Journal of Accounting Research*, 1966, pp. 71–111.
4. L. Bernstein. *Financial Statement Analysis*. Homewood, IL: Richard D. Irwin, 1978.
5. B. Buchanan and E. Shortliffe. *Rule-Based Expert Systems*. Reading, MA: Addison-Wesley, 1985.
6. W. Chudson. *The Pattern of Corporate Financial Structure: A Cross Section of Manufacturing, Mining, Trade and Construction, 1937*. New York: National Bureau of Economic Research, 1945 (as referenced in Lev [14]).
7. K. Cohen, T. Gilmore, and F. Singer. "Bank Procedures for Analyzing Business Loan Applications." In *Analytical Methods in Banking*. Homewood, IL: Richard D. Irwin, 1966.
8. D. Greene, R. Meservy, and S. Smith. "An Adaptive Learning System for Audit Selection," Unpublished paper, Carnegie Mellon University, Pittsburgh, February 1988.
9. F. Hayes-Roth, D. Waterman, and D. Lenat. *Building Expert Systems*. Reading, MA: Addison-Wesley, 1983.
10. G. Holdren. "LIFO and Ratio Analysis," *The Accounting Review*, January 1964, pp. 70–85.
11. J. Holland. *Adaptation in Natural and Artificial Systems*. Ann Arbor: University of Michigan Press, 1975.
12. J. Horrigan. "An Evaluation of Financial Ratio Analysis." Ph.D. dissertation, University of Chicago, 1967.
13. P. Keen and M. Scott Morton. *Decision Support Systems*. Reading, MA: Addison-Wesley, 1978.
14. B. Lev. *Financial Statement Analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1974.
15. J. A. Ohlson. "Financial Ratios and the Probabilistic Prediction of Bankruptcy," *Journal of Accounting Research*, Vol. 18, Spring 1980, pp. 109–131.
16. J. R. Quinlan. "Inductive Inference as a Tool for Construction of High-Performance Programs." In *Machine Learning: An Artificial Intelligence Approach, Vol I*, Michalski, R., Carbonnell, J. and Mitchell, T., eds. Tioga Press, 1986.
17. J. R. Quinlan. "The Effect of Noise on Concept Learning." In *Machine Learning: An Artificial Intelligence Approach, Volume II*, Michalski, R., Carbonnell, J. and Mitchell, T., eds. Morgan Kaufmann, 1986.
18. M. Simkowitz and R. Monroe. "A Discriminant Analysis Function for Conglomerate Targets," *Southern Journal of Business*, 1971, pp. 1–16.
19. V. Srinivasan and Y. Kim. "Decision Support for Integrated Cash Management," *Decision Support Systems*, Vol. 2, 1986, pp. 347–363.