Available online at www.sciencedirect.com

ScienceDirect

Expert Systems with Applications 36 (2009) 2130–2142

Expert Systems with Applications

www.elsevier.com/locate/eswa

The development of an ontology-based expert system for corporate financial rating

Li-Yen Shue [a], Ching-Wen Chen [a,*], Weissor Shiue [b]

[a] *Department of Information Management, National Kaohsiung First University of Science and Technology, No. 2, Juoyue Road, Nantsu, Kaohsiung 811, Taiwan*

[b] *Department of Finance, National Kaohsiung First University of Science and Technology, No. 2, Juoyue Road, Nantsu, Kaohsiung 811, Taiwan*

Abstract

This study presents a new approach in developing an expert system to assess financial quality of an enterprise. In dealing with the complexity of financial statement analysis, this approach separates knowledge content into domain knowledge of financial statements and operational knowledge of analysts' analytical process. The former represents the well defined relationships of accounting items of financial statements. The latter represents the reasoning processes that use the domain knowledge to appraise the quality of given financial statements. We apply ontology to model the content of domain knowledge, and use decision rules to represent operational knowledge. This system integrates Protégé, as domain knowledge base, and JESS, as operational knowledge base, into one complete expert system. The performance validity of the system was verified with financial statements from Taiwanese stock market.

Keywords: Ontology; Expert system; Protégé; JESS

1. Introduction

Financial statements provide historical information on the financial position and performance of a corporation, which could become important sources for investigating achievements on four primary business activities: planning, financing, investing, and operating activities. Planning activities describe a company's purpose, strategy, and tactics for activities, which could assist managers in focusing their efforts and identifying expected opportunities and obstacles. Financial activities are the means a company uses to pay for its ventures and carries out business plans, and they are also what a company must take care in acquiring and managing its financial resources. Investing activities are the acquisition and maintenance of investments by a company in selling products or providing services; a company's investments may include land, buildings, equipment, legal rights, inventories, and all components necessary for operating a company. Operating activities represent the "carrying out" of the business plan given necessary financing and investing, which is normally a mix of research, purchasing, producing, marketing, and labors. As a result, financial statements provide the main source of information for all parties who are interested in the performances of a company; including its managers, creditors, equity investors, and others. Although each of these parties may have different perspective in viewing financial statements, one major concern to all parties is the financial quality of an enterprise, which indicates the financial strength and/or weakness of its immediate future.

The primary objective of financial statement analysis is to determine the best possible estimates of and prediction about future conditions and performance. It is a complex and demanding task, because in establishing the factual insights of an enterprise, one is required to have the in-depth knowledge of accounting knowledge and practices, as well as the knowledge of operations of the particular business involved. Thus, depending on the experience and

* Corresponding author. Tel.: +886 7 6011000x4111; fax: +886 7 6011042.

E-mail address: chingwen@ccms.nkfust.edu.tw (C.-W. Chen).

0957-4174/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2007.12.044

expertise of accountants, some are more capable than others in diagnosing the true picture of the performance of a company, or discovering problems that are otherwise not obvious from the statement itself. When in analyzing a financial statement, it is well understood that even professional analysts may be subject to the constraints of personal subjective views, physical and mental fatigue, or possible environmental factors, and are not able to provide consistent appraisals. As a result, expert systems have been applied to imitate the decision processes and inferencing logics of financial experts (Matsatsinis, Doumpos, & Zopounidis, 1997; Nedovic & Devedzic, 2002; Pacheco, Martins, Barcia, & Khator, 1996). One major issue with traditional expert system methodologies lies in the fact that all knowledge is usually lumped together to form a "knowledge base". Domain knowledge and operational knowledge that use domain knowledge are mingled together and is generally called the "knowledge". Such a design may become inefficient when the domain becomes large or the decision knowledge becomes complex (Chan & Johnston, 1996; Lee & O'Keefe, 1996; Lin, Tseng, & Tsai, 2003). Another major problem associated with this design is the fact that the knowledge of one system could rarely be used by another system, which prevents the cost of knowledge establishment to be lowered by sharing with other systems (Chan & Johnston, 1996; Lee & O'Keefe, 1996).

This study presents a new approach in developing an expert system for assessing financial quality of an enterprise by separating the knowledge body into domain knowledge and operational knowledge. The former represents the well defined relationships between various accounting categories and items, and the latter refers to the analytical processes of utilizing the domain knowledge for assessment. We apply ontology to model the domain knowledge and use production rules to represent the operational knowledge. The rest of this paper is arranged as follows. Section 2 briefly presents related methodologies. Section 3 describes the analysis of domain knowledge and the development of domain ontology. Section 4 presents the development of operational knowledge in rule bases. The system framework and its performance with selected cases are given in Section 5, and Section 6 provides the conclusions of this study.

2. Literature review

This section reviews the methods for analyzing financial statements, in particular the ratio analysis, the definitions of ontology and ontology engineering in the era of knowledge management, and provides a brief description of both JESS and Protégé for the development of this system.

2.1. Analysis of financial statements

In analyzing financial statements, users can choose from a variety of tools to best suit their specific needs. In general, there are three methods for analyzing financial statements: comparative financial statement analysis, common-size financial statement analysis, and ratio analysis (Bernstein & Wild, 1998; Ross, WesterfIeld, & Jaffe, 2002; White, Sondhi, & Fried, 2003). Comparative financial statement analysis is conducted by setting all financial statements side by side for reviewing changes and trends, which is normally based on a year-to-year or multiyear values in individual categories. A comparison of statements over several years would reveal direction, speed, and extent of trends. Common-size financial statement analysis is an inquiry into the internal structure of financial statements by examining the proportion of a total group or subgroup an item represents, and it particularly focuses on sources of financing and composition of investments. This method is especially useful for inter-company comparisons because financial statements of different companies are recast in common-size format. Probably the most widely used financial analysis technique is ratio analysis. This method expresses relationships between two items on the financial statement, and analysts use related ratios together to determine credit and investment standing, to evaluate risks, trends, and to identify any peculiarities.

Given the large quantity of items in financial statements, a long list of meaningful ratios can be derived. Based on their relationship with the passage of time, financial statements can be classified into two types: static and dynamic ratios. Static ratios are calculated from balance sheet data, and present a "snapshot" of the business; current ratio and net worth to total debt ratio are typical ones. Meanwhile, some ratios are classified as dynamic, velocity, or activity ratios because they are based wholly or in part on matters of financial movement; sales to inventory ratio and sales to receivable ratio fall into this category. According to Duft (2007), most analysts, however, prefer to classify ratios according to their functions, which include categories of tests of: profitability, liquidity, leverage, and efficiency. Liquidity ratios measure a firm's ability to meet its current obligations; a business is believed to be in trouble if it is unable to meet its current obligations as they become due. Profitability ratios measure management's ability to control expenses and to earn a return on the resources committed to the business. Leverage ratios measure the degree of protection of suppliers of long-term funds and can also aid in judging a firm's ability to raise additional debt and its capacity to pay its liabilities on time. Efficiency ratios measure how effective a company is in utilizing its resources. Due to popularity of this method, our analyst suggests that we apply financial ratios in developing the evaluation module of the expert system.

2.2. Ontology and ontology engineering

The word ontology was taken from philosophy, where it means a systematic explanation of beings. In the last decade, the Knowledge Engineering community has adapted the word ontology to refer to a systematic analysis of knowledge of some domains of interest, so that it can be shared by others. The most often cited ontology definition is from Gruber (1993): "an ontology is a formal, explicit specification of a shared conceptualization". 'Conceptualization' refers to an abstract model of phenomena in the world by having identified the relevant concepts of those phenomena. 'Explicit' means that the type of concepts used, and the constraints on their use are explicitly defined. 'Formal' refers to the fact that the ontology should be machine readable. 'Shared' reflects that ontology should capture consensual knowledge accepted by the communities. While this is a very general definition, Noy, Fergerson, and Musen (2000) provide a more specific definition: "an ontology is a formal explicit representation of concepts in a domain, properties of each concept describes characteristics and attributes of the concept known as slots and constrains on these slots". Swartout, Ramesh, Knight, and Russ (1997) relates ontology to knowledge base by saying that "an ontology is a hierarchically structured set of terms to describe a domain that can be used as a skeletal foundation for a knowledge base". In actual applications, an ontology represents a set of vocabulary and related content theory (Chandrasekaran, Josephson, & Benjamins, 1999). The vocabulary consists of terms that are used for capturing the conceptualization of the domain, and the content theory refers to the identification of specific classes of objects, their properties, and their relationships that exist in the domain. Both are applied together to express entities and relationships between entities of a domain and establish a meta-information that could be shared and reused.

Ontology engineering is the approach to follow in building an ontology. Among the widely used ontology engineering methods (Bernaras, Laresgoiti, & Corera, 1996; Fernández, Gómez-Pérez, & Juristo, 1997; Gruninger & Fox, 1995; Uschold & Gruninger, 1996; Uschold & King, 1995), we decided that the TOronto Virtual Enterprise (TOVE) method proposed by Gruninger and Fox (1995) can better facilitate the development of our domain ontology of financial statements. TOVE consists of six phases: motivation scenario, informal competency question, terminology, formal competency question, axiom, and completeness theorem. Motivation scenario describes the situations that can be better dealt with ontology than with current methods. These situations are then translated into question forms in the Informal Competency Questions phase, and these questions are to be answered by the ontology system when it is completed. The terminology phase examines the terms that are used for describing various knowledge concepts of the domain, and these terms may be formally used in the system development of the Formal Competency Question phase. The Formal Competency Question phase applies a logic language to translate the questions in the Informal Competency Question, along with terminologies and their conceptual relationships, into logic statements that can be processed by computers. The Axiom phase also uses these terminologies to establish further inference processes that are otherwise difficult with the formal competency questions. The last phase is to verify the effectiveness of the completed system through questions in Competency question.

2.3. Java expert system shell (JESS)

JESS (Friedman-Hill, 1995, 2003) is a Java based expert system shell, and was developed by Sandia National Laboratories. Its former form is C Language Integrated Production System (CLIPS) (Giarratano & Riley, 2005) and was also developed by NASA earlier. With the Rete Algorithm (Forgy, 1982), JESS, and CLIPS are very efficient in pattern matching. In addition to the functions that are available in CLIPS, JESS has added other functions that make it a very powerful expert system shell (Eriksson, 2003; Friedman-Hill, 2003). Notably, JESS is capable of conducting inferences with both Forward Chaining and Backward Chaining functions, which makes it being compatible with most expert system shells. In knowledge representation, it allows users to express knowledge content in both rules as well as frames. In addition, it has greatly enhanced its modularity function that raises its capability in system maintenance and efficiency. This function is implemented through its "DefModule" function, and it allows one to modularize conceptual and/or logical segments of a system, so that a clearer structural framework can be designed. This function is especially important when a system becomes complex and has to be revised from time to time (Davis, 1990; Lee & O'Keefe, 1996; Lin et al., 2003).

2.4. Protégé

Protégé (Noy et al., 2000; Grosso et al., 1999) was developed by Medical Informatics of Stanford University, and was designed as a platform to reduce the difficulty in knowledge acquisition, which has been recognized as a major bottleneck in developing knowledge system. It is an ontology based development, which allows users to develop knowledge taxonomy and express relationships between categories with ease. One of its important features is the extensible architecture, which enables its integration with other applications; thus one can easily connect external semantic modules to Protégé. It also provides customization features to allow users with maximum flexibility in building knowledge bases. The elements of its knowledge model consist of classes, slots, facets, and instances. A class describes a category of objects or concepts that are of the same properties. Instances are the actual entities of a class. Slots represent attributes of classes and instances, and facets express additional information about slots. In addition to these features, this platform provides a set of Open Source API, so that users can design their components in JAVA as plugins to customize the knowledge base.

3. Domain knowledge of financial statements

Each of the major published financial statements is designed for different purposes. Balance sheet indicates the book value of a business by presenting the compositions of assets and liabilities at a particular date, while income statement presents revenue from sale of products and services and related expenses, and indicates how revenue is transformed into Income. From the knowledge point of view, just the final monetary figures of balance sheet do not provide the insight on how the various funds are utilized as well as the effectiveness of their utilization. This knowledge may be discovered through associations with accounting items of income statement or even cash flow statement, which records breakdowns of revenue and expenses of various transactions. An analyst must work with more than one statement to be able to trace the flows of funds and consequently determine their performances. In other words, the diagnostic knowledge as a whole must encompass all financial statements. In this research, for the sake of simplicity, we only use balance sheet and income statement to demonstrate the analysis of the domain knowledge and the development of its knowledge structure. In addition, from the system point of view, the naming practice of present accounting items is in fact embracing the unifying abstract notion of objects, where an object represents a collection of individuals with common features, which is normally called a class in system design. In the following analysis, we will apply the object oriented approach to develop the taxonomy of domain knowledge.

3.1. Relationships between accounting items

Accounting is built upon the fundamental accounting equation:

$$\text{Assets} = \text{Liabilities} + \text{Owners' Equity} \tag{1}$$

This equation must remain in balance and for that reason our modern accounting system is called a dual-entry system. This means that every transaction that is recorded in accounting records must have at least two entries; if it only has one entry the equation would necessarily be unbalanced. While the balance sheet reports assets and financing of those assets at a point in time, the income statement reports revenues and expenses over a period of time. Any two consecutive balance sheets are connected by the corresponding income statement. The relation between balance sheet and income statement can be examined by decomposing the accounting equation shown in Fig. 1. As shown, revenue and expenses are recorded to Net Income (or net loss) account, which is a temporary account of income statement. At the end of the accounting period, revenues and expenses are cleared for the new accounting period. Their balances flow into retained earnings, and then to owners' equity. From Fig. 1, one can see that any increase in expenses would incur the similar amount in liabilities, so that the accounting equation can be balanced. However, this is not the only way to keep the equation in balance; other ways include the reduction of assets, partly reduction of assets and partly increase of liabilities, and other possible combinations.

From the ontology prospective, the relationships between accounting items can be described in terms of structural relationship and semantic relationship, as shown in Fig. 2. The structural relationship represents the part-whole relationship between items in each statement. The construction of the present form of balance sheet and income statement must follow a very rigorous format as is demanded by the accounting profession. In general, items are expressed in terms of accounting categories and in the form of category tree; each category may consist of further subcategories, until it reaches the last layer. Thus, items of different layers on the tree structure do present the structural relationship. For example, the assets category of the balance sheet may have three subcategories in the subsequent layer: "current assets", "long-term assets", and "other assets". These three categories make up the components of the assets category, and each of the three categories represents one component of the assets category. This two-way part-whole structural relationship extends all the way to the line items of the last layer, and one could

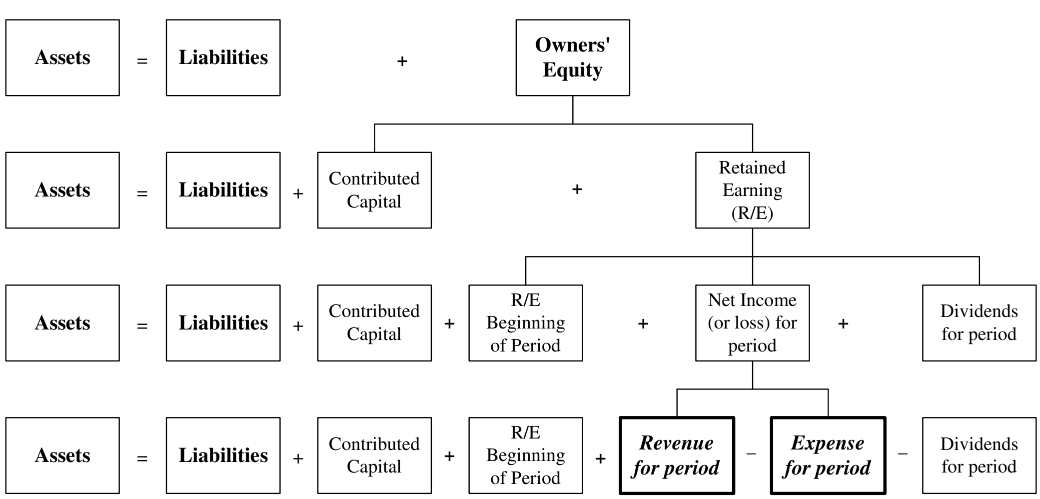

Fig. 1. Decomposition of accounting equation.

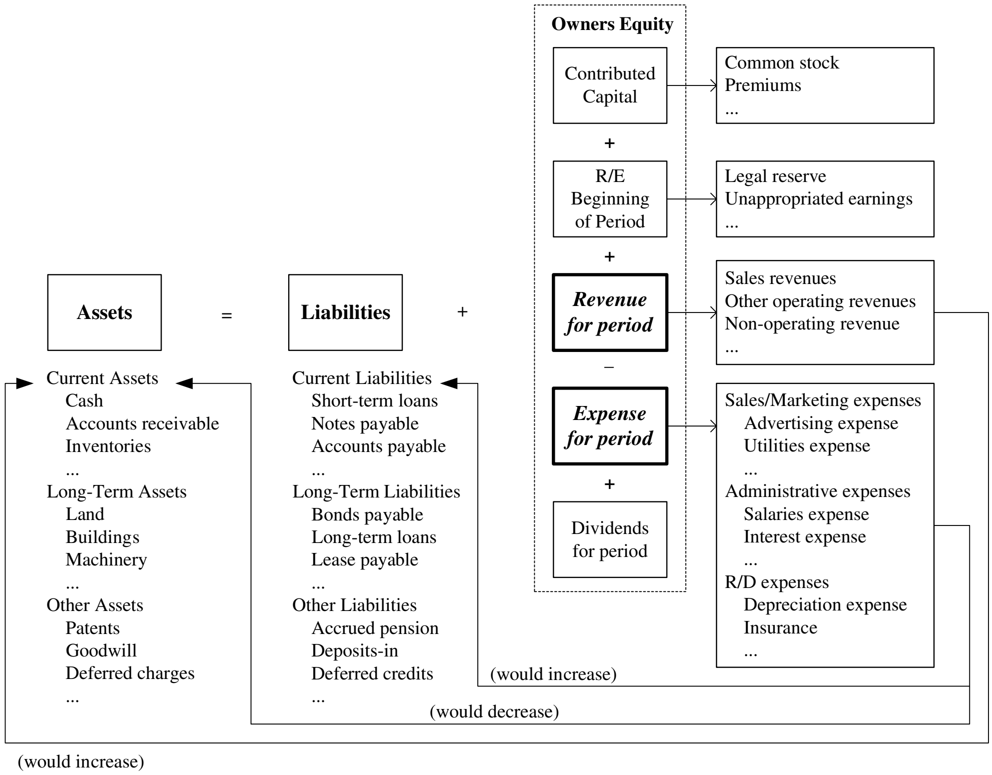

Fig. 2. The structure and semantic relations between accounts.

identify the structural relationship between two distant items through intermediate layers. In addition, the semantic relations between items represent another dimension of relationship, which basically represents the managerial concepts in performance evaluation, and is essential for the analytical purpose. One important semantic relationship is the causal relationship between items, which presents the input–output, source–outcome, and cause–effect association. The semantic relationship could relate items across different structural layers of a statement or items from different statements. For example, in balance sheet, the liabilities and owners' equity are treated as the "source" for the assets, and, in income statement, the expenditure is treated as the "cost" of operating revenue; enough through they do not have any direct structural relationship. Further, the assets of balance sheet is treated as the "source" for generating Operating_Revenue of the income statement. These semantic relationships form the basis in measuring efficiency and effectiveness, and can be further formulated into various financial ratios for measuring performances of an enterprise.

In summary, the basic structure relations could be expressed with:
hasComponent (an item is made up of certain items), and
isA (an item is a member of another item).

The semantic relationships could be as diverse as needed in expressing the concepts of diagnoses. Some of the common ones and examples are listed in Table 1.

Fig. 3 presents a simple semantic network for illustrating relationships between balance sheet and income statement through both structural and semantic relationships. This figure indicates that the class "balance sheet" has three components (hasComponent): Assets, liabilities, and owners' equity, and the "income statement" also has three components (hasComponent): operating revenue, expenditure, and surplus. With each component being a unique concep-

Table 1
Some of the basic semantic relations and corresponding examples

Semantic relations	Examples
isSumOf (the value of an item is the sum of values of other items)	Current asset is the sum of cash, account receivable, inventory, and other related accounts
isCostFor (an item is considered as a cost of another item)	Depreciation is the cost for other long-term asset like buildings
isSourceFor (an item is considered as the source of another item)	Revenue is the source of cash or account receivable
hasDenominator (an item serves as the denominator of a ratio)	Total asset is the denominator of the equity ratio
hasNumerator (an item serves as the numerator of a ratio)	Net income is the numerator of return on sale

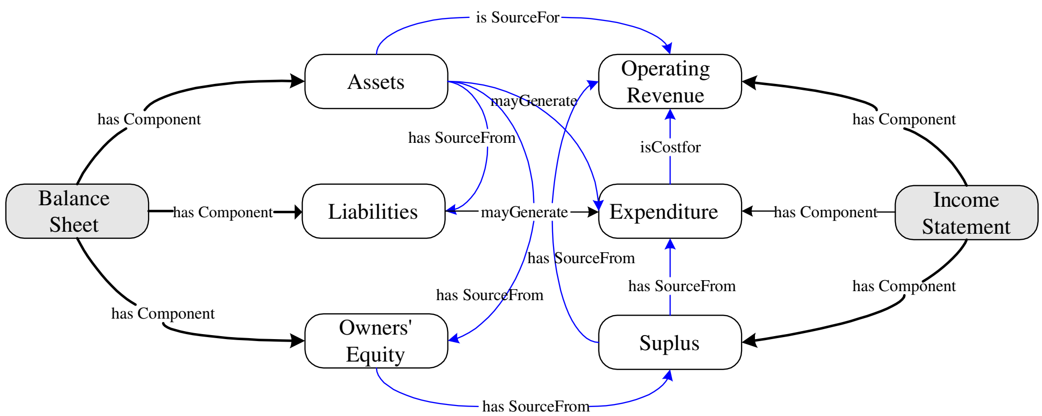

Fig. 3. A simplified relationships between balance sheet and income statement.

tual grouping, they are treated as classes. Between the two statements, the assets class has sources from (hasSourceFrom) liabilities and owners' equity. The assets class is also the source for (isSourceFor) generating operating revenue. The expenditure class is the cost for (isCostFor) generating operating revenue, keeping liabilities, or decreasing assets. The surplus class is calculated from operating revenue and expenditure and thus has sources from (hasSourceFrom) them.

3.2. Domain knowledge structure with financial ratios

In addition to the relationships of items of financial statements, the relationships that represent concept of financial ratios must be included in the knowledge structure as well, because they are the basis for conducting the financial diagnosis. Our analyst, who is a certificated public accountant (CPA) and has been in his profession at Taiwan for more than 20 years (Li, Shue, & Shiue, 2000), indicates that there are four categories of financial ratios that are effective in assessing financial quality: short-term repayment ability, long-term repayment ability, operation efficiency, and profitability. Each category may consist of more than one ratio from both statements, and all together there are twelve ratios to be considered. These ratios are well defined in accounting profession, and they are listed as follows (Bernstein & Wild, 1998; Ross et al., 2002; White et al., 2003):

1. Short-term repayment ability:
   (1) Current ratio = Current assets/current liabilities.
   (2) Net operating cycle (or cash conversion cycle) = Average receivables collection period + average processing time for inventory − payables payment period.
2. Long-term repayment ability:
   (1) Equity ratio = Total equity/total assets.
   (2) Time interest earned = (Net profit before taxes + interest paid)/interest paid.
   (3) Long-term funds to fixed assets = (Owners' equity + long-term interest − bearing liabilities)/net fixed assets.
3. Operation efficiency:
   (1) Total assets turnover (times) = Net sales/average total assets.
   (2) Fixed assets turnover (times) = Net sales/average fixed assets.
   (3) Accounts receivable turnover = Net credit sales/average accounts receivable.
   (4) Inventory turnover (times) = Cost of goods sold/average inventory.
4. Profitability:
   (1) Return on sale = Net income/net sales.
   (2) Return on equity = Net income/average owners' equity.
   (3) Return on total assets = {Net income + interest expense × (1 − tax rate)}/average total assets.

We encompass these ratio concepts into our domain knowledge structure. Each ratio is calculated with values of two items, which are presented as two slots of the ratio class; one is (hasNumerator) and the other is (hasDenominator). For example, total assets turnover can be obtained by dividing the sale revenue in income statement from total assets at balance sheet. Fig. 4 presents a detailed semantic network of the twelve ratio classes in conjunction with relevant classes of the two statements. In the figure, we follow the convention of Protégé to name classes as either abstract or concrete class; the former may have instances while the latter may not.

3.3. Domain ontology

With the knowledge structure that consists of taxonomical concepts of accounting categories and financial ratios, we apply ontology to model the domain knowledge and represent the knowledge content in terms of classes, relations, attributes, and individuals. In developing the domain ontology, we adopt TOronto Virtue Enterprise (TOVE)

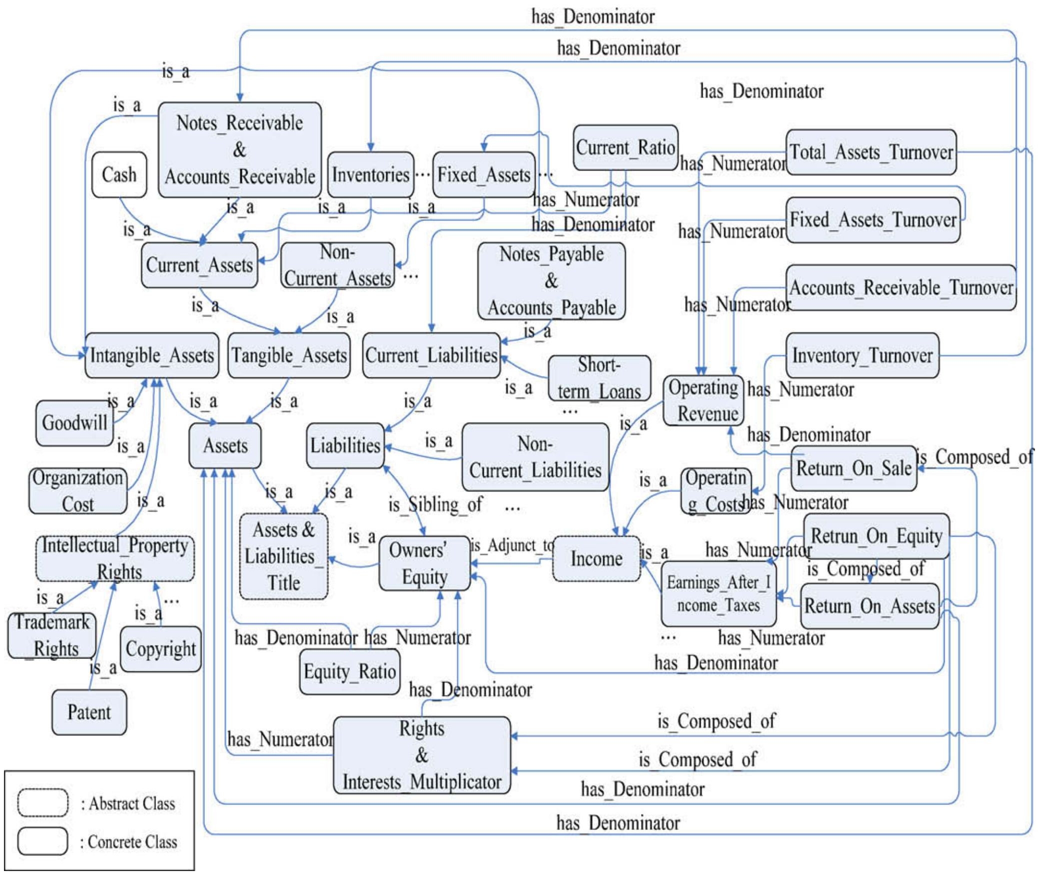

Fig. 4. A simplified semantic network of domain knowledge with financial ratios.

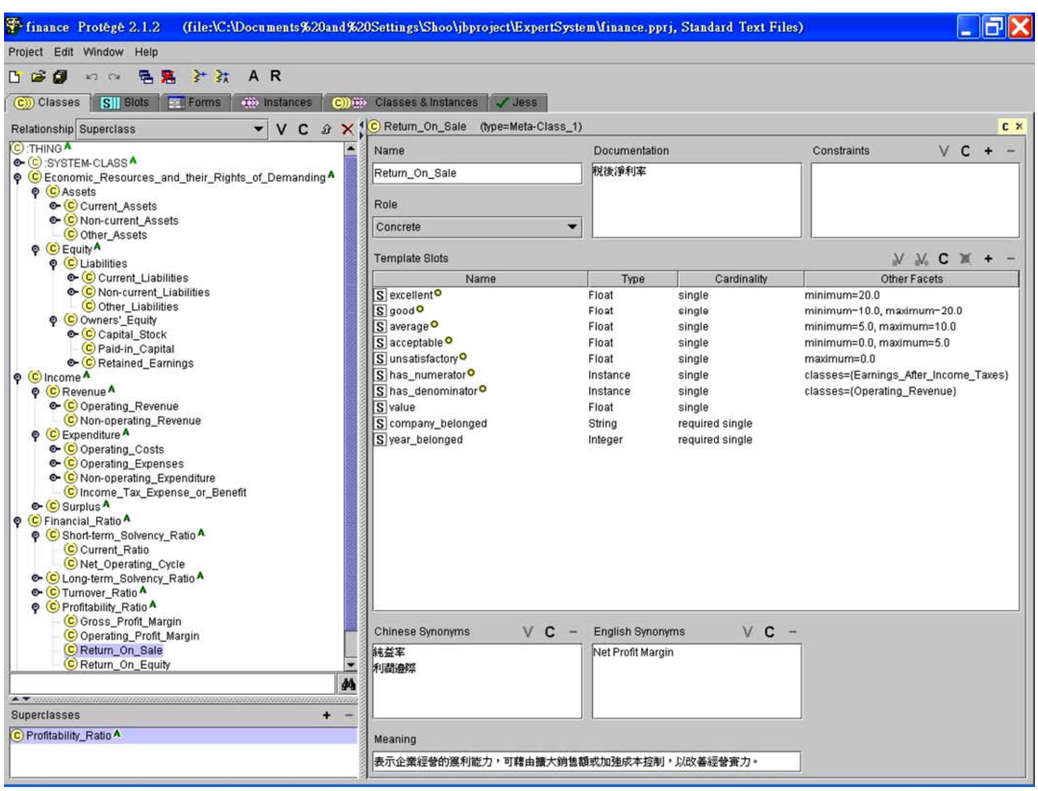

Fig. 5. Class taxonomy on Protégé.

(Gruninger & Fox, 1995) Ontology Engineering approach. This approach consists of six phases: motivation scenario, informal competency question, terminology, formal competency question, axioms, and completeness theorem. The six phases encompass the identification of problem domain, the analysis of knowledge content, and the development of relationships and related reasoning processes. Since the major reasoning part of the completed system, which is the accountant's diagnostic process, will be handled by the rule-based module, we hence skip the development of reasoning process, which includes formal competency question and axioms. Despite of this, the domain ontology must be so designed that the completed system can properly answer the intended questions. These questions are initially expressed in the motivation scenario and later become informal competency questions. For this study, with given financial statements of an enterprise, our informal competency questions consist of:

1. What is the rating of overall financial performance and the explanation for the rating?
2. What is the rating of short-term repayment ability and the explanation for the rating?
3. What is the rating of long-term repayment ability and the explanation for the rating?
4. What is the rating of operation efficiency and the explanation for the rating?
5. What is the rating of profitability and the explanation for the rating?

Fig. 5 presents the complete class structure with Protégé. The first layer of the structure consists of three classes: economic resources and their right of demanding, income, and financial ratios. As shown, each class has its own slots (attributes) and relationships with other class, and may consist of more than one subclass. For example, the class of financial ratios has four subclasses; among them, the profitability ratio class consists of four further subclasses. One of them is return on sales, which has its numerator from class earnings after income tax and its denominator from class operating revenue.

4. Knowledge representation of operational knowledge

Operational knowledge is the diagnostic knowledge, which takes domain knowledge of financial statements and infers a conclusion on the financial quality of a company. Normally, the operational knowledge consists of a series of analytical steps while examining the items of financial statements. In general, an analyst would examine some major items to start with, and exercises his/her professional judgments to determine if a conclusion can be made, or other relevant items are needed for further verification in order to reach a conclusion. There may be many intermittent steps before a final conclusion can be reached. An in-depth discussion with our analyst has led us to apply rule-based methodology to represent his operational knowledge. In the following, we use decision rules to develop the operational knowledge provided by our accountant.

The complete operational knowledge is made up of decision rules for each of the four categories of financial ratios and the overall decision. We follow the general convention in financial analysis to evaluate each rating into one of the five quality grades: 5 for being excellent, 4 for being good, 3 for being above average, 2 for being acceptable, and 1 for being unsatisfactory. The decision process starts with the evaluation of financial ratios pertaining to each decision criterion, the evaluation of each decision criterion by combining ratings of its ratios, and the overall evaluation by combining ratings of the four criterions. Fig. 6 presents an example of decision rules, which take input from financial statements and evaluate rating for each of the four ratios of operating efficiency: Total_Assets_Turnover (TAT), Accounts_Receivable_Turnover (ART), Inventory_Turnover (IT), and Fixed_Assets_Turnover (FAT). Fig. 7 presents the decision rules for evaluating the four criterions and the overall rating for one actual case. In the figure, the inputs are ratings of ratios of each of the four decision criterions, which lead to the criterion ratings: short-term liquidity = good, long-term solvency = average, operating efficiency = good, profitability = average. The overall assessment of the company's financial quality is based on these ratings, and it is "average" in this case. Based on the combinations of number of ratios for each criterion, four criterions, and five grades for each rating, there are more than 1600 decision rules in the operational knowledge.

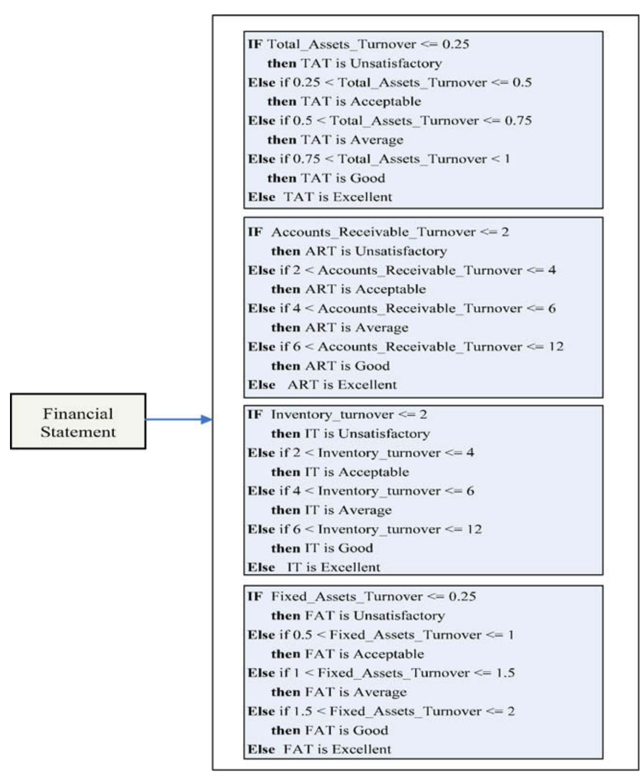

Fig. 6. Decision rules for rating four ratios of operating efficiency.

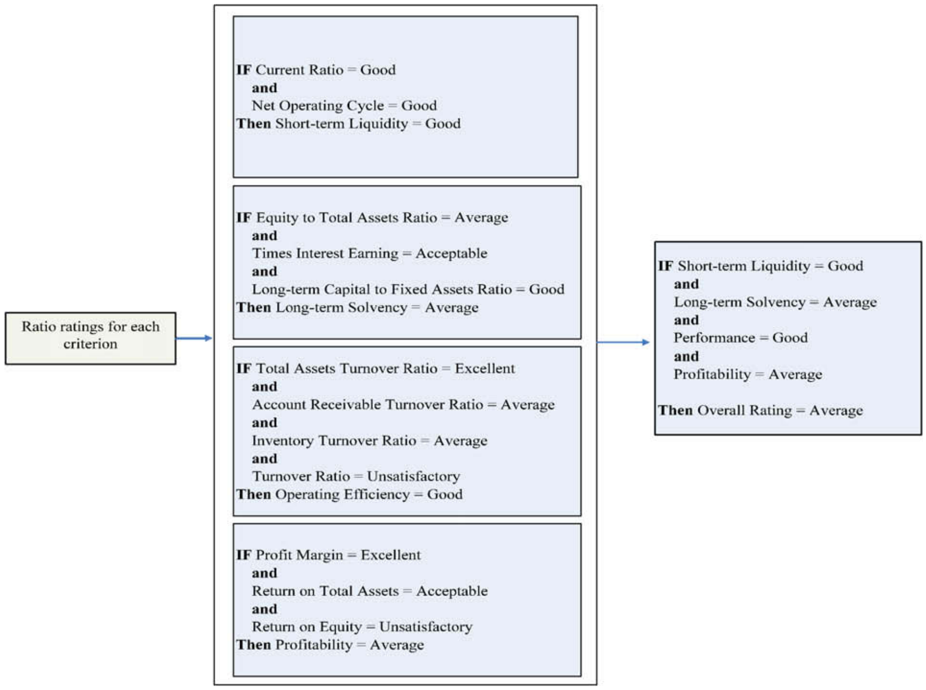

Fig. 7. An example of decision rules for criterion and overall rating.

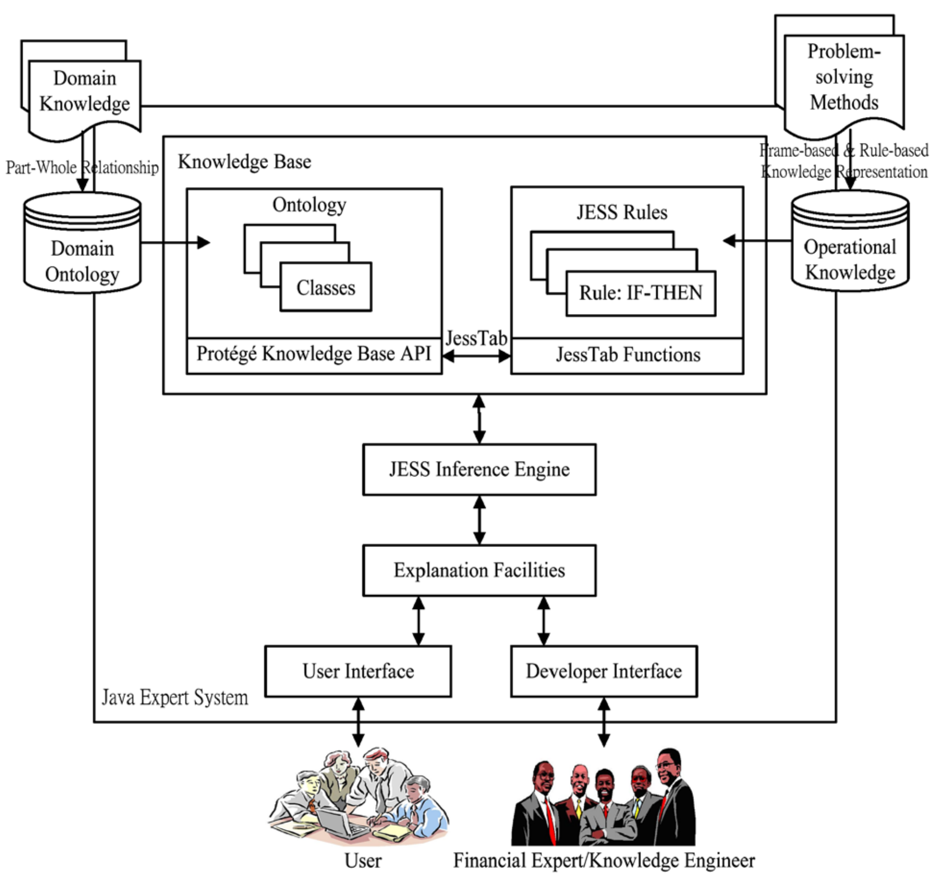

Fig. 8. The system framework.

```
import jess.*;
import JessTab.*;
...
Rete engine = new Rete();
engine.addUserpackage(new JessTabFunctions());
engine.executeCommand("(load-project finance.pprj)");
engine.executeCommand("(mapclass Income reactive)");
... ...
```

Fig. 9. Commands to bridge together JESS and Protégé.

5. The system framework

The proposed system development framework is shown in Fig. 8. This framework is made up of modules of knowledge base, inference engine, explanation facility, user interface, and developer interface. The knowledge base module is the core of the system; it consists of domain knowledge of financial statements and operational knowledge of analysts' decision rules. The user interface accepts the original facts from financial statements and invokes the inference engine to activate the decision rules in the operational knowledge base. The decision rules will then access relevant data residing in the domain knowledge, which may in term initiate more decision rules. The final conclusion of assessments will also be presented through the user interface, along with relevant explanation from the explanation module. The developer interface allows system developers to modify the contents of the system. The entire framework is built in JAVA environment, which is essential in achieving integration between modules during execution.

In developing the knowledge base module, we made use of the well known ontology-based knowledge platform Protégé to build the domain knowledge, and we utilize the rule facility of JESS to build the operational knowledge. Protégé is JAVA based and is designed with an open architecture, which makes it possible to communicate with other JAVA based systems through Plug-ins. For the case with JESS, the Plug-in JessTab (Eriksson, 2003) was specifically designed to serve as a bridge to allow Protégé to communicate with JESS. API of Protégé is able to work with functions of JessTab, which enables JESS's inference engine to function as the inference engine of the combined expert system. During execution, functions of JessTab can manage Protégé's knowledge base, and allows rules in JESS to access contents of the domain ontology in Protégé. This ability enhances the level of reasoning complexity of Protégé beyond its original capability. Fig. 9 presents commands for actual implementation, which brings both Protégé and JESS into the JAVA environment. The "mapclass" command maps the contents, including instances, attributes, and corresponding relational structure, of Protégé into a template that is accessible by the JESS engine.

Table 2
Extracts from official financial statements of 2003

	Mediatek Incorporation	Ritek Corporation
Notes and accounts receivable	3,706,372	4,728,886
Notes and accounts payable	4,181,360	2,699,966
Net sales	38,064,419	4,319,832
Inventories	2,075,508	3,114,072
Current liabilities	5,887,266	9,103,215
Operating expense	18,409,445	4,459,096
Current assets	29,198,229	15,573,544
Long-term liabilities	10,643	16,788,737
Interest expense	44	25,385
Property, plant and equipment	1,048,591	30,401,794
Total liabilities	5,952,229	26,622,775
Income before income tax	16,609,044	−633,692
Total assets	41,987,303	76,346,034
Total shareholders' equity	36,035,074	49,723,258
Net income	16,522,089	−469,843

Amounts in thousands of New Taiwan dollars (NT$).

6. Performance verification

The Taiwan Stock Exchange Corporation[1] (TSEC) was incorporated in October 1961 under the jurisdiction of the Financial Supervisory Commission (FSC) of Taiwan. At least one-third of board directors and supervisors of TSEC are appointed by the securities authority to represent the public interest. TSEC has the primary function of a centre for trading listed securities and providing the essential systems necessary to facilitate securities trading. To keep the financial information openly and correctly, a TSEC listed company shall select a professional, responsible and independent CPA to be its external auditor, who shall perform regular reviews of the financial conditions and internal control measures of the company. Thus, the listing company must submit its annual and semiannual CPA-audited financial reports, as well as the first and third quarter CPA-reviewed financial reports to TSEC for examination.

[1] URL: http://www.tse.com.tw/en/.

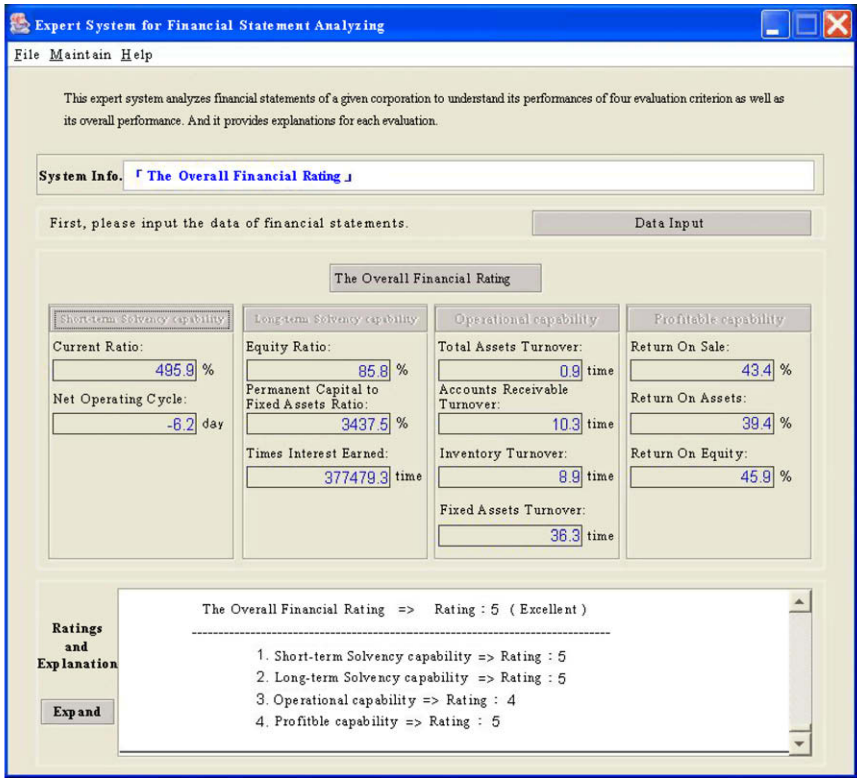

Fig. 10. Financial ratings for Mediatek Incorporation.

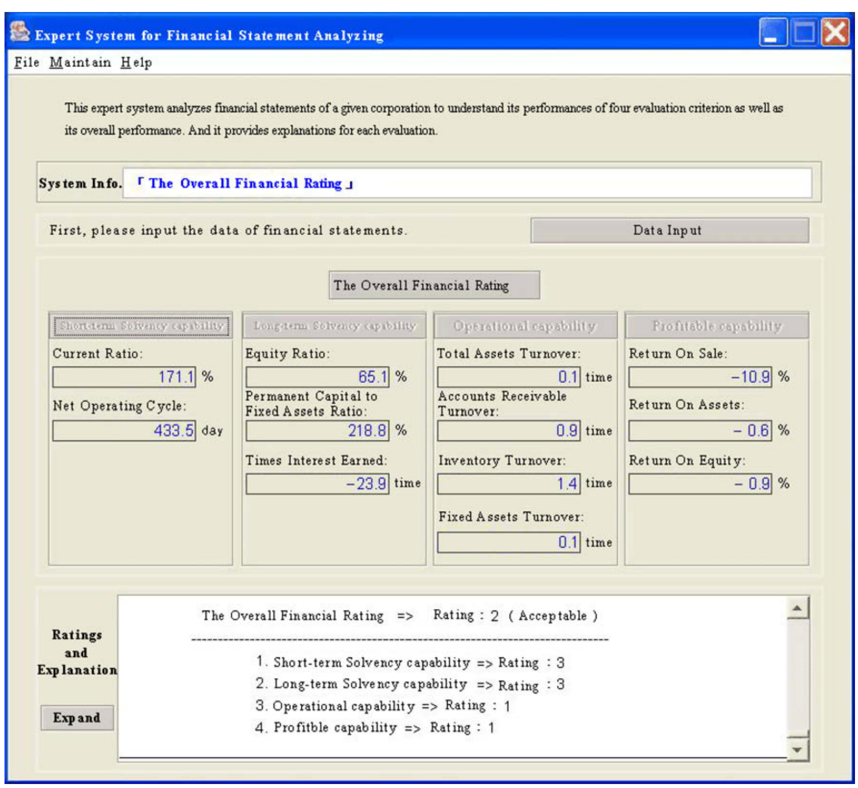

Fig. 11. Financial ratings for Ritek Corporation.

In order to verify the validity of the system, we run the system with financial data of several companies that are listed on Taiwan Stock Exchange Corporation, and compare its rating results with that of our accountant. The following presents two of them. The two chosen companies are Ritek Corporation with stock code 2349 and Mediatek

Incorporation with stock code 2454 at Taiwan Stock Exchange Corporation (TSEC). Ritek Corporation was established in 1988 in Hsinchu Industrial Park of Taiwan and is specialized in storage media. Since its establishment, it had produced Taiwan's first CD, DVD, and OLED. Over the years, it has become the largest manufacturer of optical storage media in the world, and was the first optical disc manufacturer in the world that can simultaneously mass produce CD-R/-RW, DVD-R/-RW, DVD+R/+RW, DVD-RAM and HD DVD. Lately, it has devoted to the development of flash memory and consumer electronic products. Mediatek Incorporation was also established in Hsinchu Scientific Park of Taiwan in 1997 and is a fables IC company. It is dedicated to the development of comprehensive digital media integrated chipset solution, and is a leading digital media solution provider. Since its establishment, MediaTek has emphasized very much on R&D, and has become a worldwide leader in high-end consumer IC solutions.

Both companies, as described, are of typical high-tech companies and sell their products worldwide; however, as shown in the following table, their performances differ noticeably in 2003. Their financial statements can be accessed through Taiwan Stock Exchange Corporation (TSEC) or major Taiwanese financial publications. Figures in Table 2 are extracted from their official financial statements of 2003, which are available from TSEC's Market Observation Post System[2] (MOPS) with codes 2349 for Ritek and 2454 for Mediatek, respectively.

With data from these statements, our system produces assessment results that are shown in Figs. 10 and 11 for both companies separately. It is worth noting that, from Fig. 10, the index of the net operating cycle (NOC) for Mediatek is −6.2 days. Net operating cycle is also known as cash conversion cycle and is calculated with the formula:

Net operating cycle = Average receivables collection period + average processing time for inventory − payables payment period.

The success of Mediatek has earned itself remarkable credit from its business partners, and this has led to the increase of the payable payment period for Mediatek, the reduction of the average receivables collection period, and the reduction of average processing time for inventory. These effects lead to a negative net operating cycle that is excellent in practice. Similarly, it is also worth noting that the times interest earned (TIE) for Ritek, Fig. 11, is −23.9 times. With the formula for calculating TIE as:

Time interest earned = (Net profit before taxes + interest paid)/interest paid.

One could be sure that Ritek must have a loss before income tax, which is (NT$ 633,692,000) as shown in Table 2. In addition, we can see that Mediatek's profitability performance is much better than that of Ritek, in terms of return of sales (ROS), return of assets (ROA), and return of equity (ROE) respectively. From Figs. 10 and 11, we can see that Mediatek's performance is much superior to that of Ritek in all four categories of financial ratios. The comparisons are summarized in Table 3.

Table 3
Financial rating for Mediatek and Ritek

Categories of financial ratios	Mediatek	Ritek
Short-term solvency capability	5	3
Long-term solvency capability	5	3
Operational capability	4	1
Profitability performance	5	1
Overall rating	5	2

In both cases, the assessments are in agreement with our accountant's own assessments. As shown, the overall rating for Mediatek is 5, at the top of the scale, and is excellent. This is also reinforced by its market performance; it has since become one of the few very successful stocks on the Taiwan stock market, and its price has increased many folds. On the other hand, the overall rating for Ritek is only 2, at the lower end of the scale, and is barely acceptable to the market. This stock has in fact being treated as a mediocre stock by the Taiwan Stock Exchange Corporation (TSEC) in the past three years.

7. Conclusions

We separate the knowledge of financial statements analysis into domain knowledge and operational knowledge to develop an expert system for assessing financial quality of an enterprise. The former represents the well defined relationships of accounting items of financial statements, and the later represents the analytical processes of the analyst during the diagnostic process. The domain knowledge including financial ratios is expressed in terms of structural and semantic relationships. We apply ontology to develop the knowledge taxonomy of the domain knowledge, and apply production rules to represent operational knowledge of the analyst. The complete system consists of Protégé module that serves as the domain knowledge base, JESS module that serves as the operational knowledge base, and the user interface module that is developed in JAVA. The application of JessTab plugin enables the search engine of JESS to assess the contents of instances of Protégé including the relations of slots, and function as the inference engine of the integrated system. The performances of this system were validated through its application to actual financial statements of several listed companies of Taiwan stock market.

For future study, we feel it is worthwhile to develop an overall knowledge structure, which could integrate all financial statements that may be set up independently as individual ontology.

[2] URL: http://emops.tse.com.tw/emops_all.htm.

References

Bernaras, A., Laresgoiti, I., & Corera, J. (1996). Building and reusing ontologies for electrical network applications. In *Proceedings of the 12th European conference on artificial intelligence (ECAI_96 in Budapest, Hungary)* (pp. 298–302).

Bernstein, L., & Wild, J. (1998). *Financial statement analysis: Theory, application, and interpretation*. Boston: Irwin McGraw Hill.

Chan, C. W., & Johnston, M. (1996). Knowledge modelling for constructing an expert system to support reforestation decisions. *Knowledge-Based Systems, 9*, 41–59.

Chandrasekaran, B., Josephson, J. R., & Benjamins, V. R. (1999). What are ontologies, and why do we need them?. *IEEE Intelligent Systems 14*(1), 20–26.

Davis, J. S. (1990). Effect of modularity on maintainability of rule-based systems. *International Journal of Man–Machine Studies, 32*, 439–447.

Duft, K. D. (2007). Financial ratio analysis: An aid to agribusiness management. <http://classes.ses.wsu.edu/EconS452/section-Duft/docs/FinRatio1.pdf> Accessed 17.02.2007.

Eriksson, H. (2003). Using JessTab to integrate Protégé and jess. *IEEE Intelligent Systems, 18*(2), 43–50.

Fernández, M., Gómez-Pérez, A., & Juristo, N. (1997). *Methontology: From ontological art towards ontological engineering*. Stanford: Spring Symposium Series.

Forgy, C. L. (1982). Rete: A fast algorithm for the many pattern/many object pattern match problem. *Artificial Intelligence, 19*, 17–37.

Friedman-Hill, E. J. (1995). Jess, the rule engine for the java platform. <http://herzberg.ca.sandia.gov/jess/> Assessed 21.11.2003.

Friedman-Hill, E. J. (2003). *Jess in action – Rule-based systems in java*. Greenwich: Manning Publications.

Giarratano, J., & Riley, G. (2005). *Expert systems principles and programming*. Boston: Thomson Publications.

Grosso, W. E., Eriksson, H., Fergerson, R. W., Gennari, J. H., Tu, S. W., & Musen, M. A. (1999). Knowledge modeling at the millennium: The design and evolution of Protege-2000. Technical Report SMI-1999-0801. Stanford, CA: Stanford University, Institute for Medical Informatics.

Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition, 5*(2), 199–220.

Gruninger, M., & Fox, M. S. (1995). Methodology for the design and evaluation of ontologies. In *Workshop on basic ontological issues in knowledge sharing (IJCAI-95), Montreal, Canada*.

Lee, S., & O'Keefe, R. M. (1996). The effect of knowledge representation schemes on maintainability of knowledge-based systems. *IEEE Transactions on Knowledge and Data Engineering, 8*(1), 173–178.

Li, S.-T., Shue, L.-Y., & Shiue, W. (2000). The development of a decision model for liquidity analysis. *Expert Systems with Applications, 19*(4), 271–278.

Lin, Y. T., Tseng, S. S., & Tsai, C. F. (2003). Design and implementation of new object-oriented rule based management system. *Expert Systems with Applications, 25*(3), 369–385.

Matsatsinis, N. F., Doumpos, M., & Zopounidis, C. (1997). Knowledge acquisition and representation for expert systems in the field of financial analysis. *Expert Systems with Applications, 12*(2), 247–262.

Nedovic, L., & Devedzic, V. (2002). Expert systems in finance – A cross-section of the field. *Expert Systems with Applications, 23*(1), 49–66.

Noy, N. F., Fergerson, R. W., & Musen, M. A. (2000). The knowledge model of Protege-2000: Combining interoperability and flexibility. In *Proceedings of the 12th international conference on knowledge engineering and knowledge management (EKAW'2000), France*.

Pacheco, R., Martins, A., Barcia, R. M., & Khator, S. (1996). A hybrid intelligent system applied to financial statement analysis. In *Proceeding of fifth international conference on fuzzy systems (FUZZ-96-IEEE), New Orleans* (pp. 1007–1012).

Ross, S., WesterfIeld, R., & Jaffe, J. (2002). *Corporate finance* (6th ed.). New York: McGraw-Hill.

Swartout, B., Ramesh, P., Knight, K., & Russ, T. (1997). Toward distributed use of large-scale ontologies. In *Proceeding of 10th knowledge acquisition for knowledge-based system workshop (in Alberta, Canada)* (pp. 9–14).

Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, methods and applications. *The Knowledge Engineering Review, 11*(2), 93–136.

Uschold, M., & King, M. (1995). Towards a methodology for building ontologies. In *IJCAI95 workshop on basic ontological issues in knowledge sharing, Montreal, Canada*.

White, G. I., Sondhi, A. C., & Fried, D. (2003). *The analysis and use of financial statements*. New York: John Wiley & Sons Inc.