MIS Quarterly

SPECIAL ISSUE: BUSINESS INTELLIGENCE RESEARCH

METAFRAUD: A META-LEARNING FRAMEWORK FOR DETECTING FINANCIAL FRAUD[1]

Ahmed Abbasi
McIntire School of Commerce, University of Virginia,
Charlottesville, VA 22908 U.S.A. {abbasi@comm.virginia.edu}

Conan Albrecht, Anthony Vance, and James Hansen
Information Systems Department, Marriott School of Management, Brigham Young University,
Provo, UT 84606 U.S.A. {ca@byu.edu} {anthony@vance.name} {james_hansen@byu.edu}

Financial fraud can have serious ramifications for the long-term sustainability of an organization, as well as adverse effects on its employees and investors, and on the economy as a whole. Several of the largest bankruptcies in U.S. history involved firms that engaged in major fraud. Accordingly, there has been considerable emphasis on the development of automated approaches for detecting financial fraud. However, most methods have yielded performance results that are less than ideal. In consequence, financial fraud detection continues as an important challenge for business intelligence technologies.

In light of the need for more robust identification methods, we use a design science approach to develop MetaFraud, a novel meta-learning framework for enhanced financial fraud detection. To evaluate the proposed framework, a series of experiments are conducted on a test bed encompassing thousands of legitimate and fraudulent firms. The results reveal that each component of the framework significantly contributes to its overall effectiveness. Additional experiments demonstrate the effectiveness of the meta-learning framework over state-of-the-art financial fraud detection methods. Moreover, the MetaFraud framework generates confidence scores associated with each prediction that can facilitate unprecedented financial fraud detection performance and serve as a useful decision-making aid. The results have important implications for several stakeholder groups, including compliance officers, investors, audit firms, and regulators.

Keywords: Fraud detection, financial statement fraud, feature construction, meta-learning, business intelligence, design science

Introduction

Financial fraud has important implications for investors, regulators, auditors, and the general public. In recent years, several high-profile frauds have been uncovered. These frauds have had a significant negative impact on the economy and stock markets worldwide, and have resulted in a general loss of confidence in the integrity of businesses (Albrecht, Albrecht, and Albrecht 2008; Carson 2003). For instance, major frauds at Enron, WorldCom, and several other firms were the principal catalyst of a 78 percent drop in the NASDAQ Index between 2000 and 2002. Despite the resulting changes to basic accounting and internal control procedures, the problem has not abated, as the number of known

[1] Hsinchun Chen, Roger Chiang, and Veda Storey were the accepting senior editors for this paper. Gediminas Adomavicius served as the associate editor.

The appendix for this paper is located in the “Online Supplements” section of the *MIS Quarterly*’s website (http://www.misq.org).

Table 1. Largest Bankruptcies in U.S. History

Company	Assets (Billions)	When Filed	Fraud Involved?
1. Lehman Brothers, Inc.	**$691.0**	**September 2008**	**Yes**
2. Washington Mutual, Inc.	$327.9	September 2008	Not yet determined
3. WorldCom, Inc.	**$103.9**	**July 2002**	**Yes**
4. General Motors Corp.	$91.0	June 2009	Not yet determined
5. CIT Group, Inc.	$80.4	November 2009	Not yet determined
6 Enron Corp.	**$65.5**	**December 2001**	**Yes**
7. Conseco, Inc.	**$61.4**	**December 2002**	**Yes**
8. Chrysler, LLC	$39.3	April 2009	Not yet determined
9. Thornburg Mortgage, Inc.	$36.5	May 2009	Not yet determined
10. Pacific Gas & Electric Co.	$36.2	April 2001	No

Source: Bankruptcydata.com (2010)

frauds in the United States has increased considerably in the past 10 years (Dechow et al. 2011). According to the Association of Certified Fraud Examiners (2010), the typical organization loses an estimated 5 percent of its total annual revenue to fraud. Moreover, many of the fraud schemes responsible for these losses span several years, underscoring the ineffectiveness of existing fraud prevention mechanisms.

Financial fraud has also contributed to the bankruptcy of major organizations. Table 1 shows that 4 of the 10 largest bankruptcies in U.S. history were associated with major financial frauds (in bold). Three other major bankruptcies, Refco, Global Crossing, and Adelphia (the 13th, 15th, and 21st largest bankruptcies), were also associated with financial frauds. There has not been sufficient time to know whether some of the other recent bankruptcies involved fraudulent activities.

Moreover, the problem appears to have global reach. For example, major incidents in Australia (Harris Scarfe and HIH), Italy (Parmalat), France (Vivendi), the Netherlands (Royal Ahold), Korea, (SKGlobal), China (YGX), Japan (Livedoor Co.), and India (Satyam) indicate that financial fraud is occurring not just in the United States, but throughout the world (Albrecht, Albrecht, and Albrecht 2008). The Association of Certified Fraud Examiners estimates that every year, worldwide financial fraud losses exceed a trillion dollars. This estimate is based on hundreds of documented financial fraud cases every year in Europe, Asia, Africa, South America, and Oceania, coupled with the reality that many frauds (and resulting losses) go undetected.

In addition to the large-scale economic implications, three stakeholders in particular stand to benefit from enhanced financial fraud detection capabilities: investors, audit firms, and government regulators (Cecchini et al. 2010; Lin et al. 2003). Investors—a term we use broadly to include individuals, investment firms, rating agencies, and others—often have little inside information on potential fraud risk. Fraud detection tools could allow investors to make better informed decisions (Albrecht, Albrecht, and Dunn 2001). The second group, audit firms, could benefit by using a fraud risk assessment during the client acceptance period, as well as during routine audits (Albrecht, Albrecht, and Albrecht 2008; Cecchini et al. 2010). Finally, enhanced fraud detection tools could help a third group, government regulators. Since regulators are typically limited in terms of their available time and resources, effective fraud detection tools could allow them to better prioritize and focus their investigatory efforts (Albrecht, Albrecht, and Dunn 2001; Dechow et al. 2011).

Despite these needs, existing methods for financial fraud detection have been unable to provide adequate fraud detection capabilities, with most studies on U.S. firms attaining detection rates of less than 70 percent (Cecchini et al. 2010; Green and Choi 1997; Lin et al. 2003). Moreover, many prior studies utilized internal (i.e., non-public) data, which is more costly and time-consuming to acquire, and is generally unavailable to many of the aforementioned stakeholders (Cecchini et al. 2010). For instance, even audit firms and regulators do not have access to internal data until an audit/investigation is already taking place (Albrecht, Albrecht, and Albrecht 2008). A need remains for techniques capable of providing enhanced detection of financial fraud using publicly available information. While financial fraud can never be confirmed without a full investigation, advances in fraud detection methods may provide red flags that warn stakeholders of the likelihood of fraud (Bay et al. 2006).

Recent developments in business intelligence (BI) technologies have elevated the potential for discovering patterns associated with complex problem domains (Watson and Wixom 2007), such as fraud (Bolton and Hand 2002). Broadly, BI technologies facilitate historical, current, and predictive views of business operations (Shmueli et al. 2007), and may suggest innovative and robust methods for predicting the occurrence of fraud (Anderson-Lehman et al. 2004). In fact, fraud detection is recognized as an important application area for predictive BI technologies (Brachman et al. 1996; Michalewicz et al. 2007). Since BI tools facilitate an improved understanding of organizations' internal and external environments (Chung et al. 2005), enhanced financial fraud detection methods could greatly benefit the aforementioned stakeholder groups: investors, audit firms, and regulators. The research objective of this study is *to develop a business intelligence framework for detecting financial fraud using publicly available information with demonstratively better performance than that achieved by previous efforts.*

To achieve this objective, we adopted the design science paradigm to guide the development of the IT artifact, the MetaFraud framework (Hevner et al. 2004). In doing so, we selected meta-learning as the kernel theory to guide the design of the IT artifact. *Meta-learning* is a specialized form of machine learning that is able to learn about the learning process itself to increase the quality of results obtained (Brazdil et al. 2008). This ability was especially useful in our context because of the complexities and nuances associated with fraud detection (Abbasi et al. 2010; Virdhagriswaran and Dakin 2006). The MetaFraud framework encompasses four components that advocate the use of a rich set of measures derived from publicly available financial statements, coupled with robust classification mechanisms. We rigorously evaluated our framework in a series of experiments that demonstrate the utility of each individual component of the framework, as well as the framework's overall effectiveness in comparison to existing state-of-the-art financial fraud detection methods.

The research contribution of this paper is the MetaFraud framework, which demonstrates that BI techniques (Shmueli et al. 2010; Vercellis 2009) based on meta-learning can be integrated via a design science artifact to detect financial fraud with substantially higher performance than that obtained by previous research. Important aspects of the architecture and process of MetaFraud include (1) robust feature construction, which includes organizational and industry contextual information and the use of quarterly and annual data, and (2) a method of fusing stacked generalization, semi-supervised learning, and adaptive/active learning. We demonstrate in the paper how these two facets of the framework substantially enhance fraud detection capabilities over existing techniques. Further, we provide (3) a confidence-level measure that identified a large subset of fraud cases at over 90 percent legitimate and fraud recall, making fraud detection using public information practicable for various stakeholders.

The remainder of this paper is organized as follows. The next section reviews previous efforts to identify financial fraud, and shows the need for more robust methods. The subsequent section introduces the meta-learning kernel theory, and describes its usefulness in the context of financial fraud. It also introduces six hypotheses by which the design artifact was evaluated. The design of the MetaFraud framework is then outlined, and details of the financial measures derived from publicly available information used to detect fraud are presented. The fifth section describes the experiments used to evaluate the MetaFraud framework, and reports the results of the hypothesis testing. A discussion of the experimental results and their implications follows. Finally, we offer our conclusions.

Literature Review

While prior financial fraud detection research has occasionally utilized internal data (e.g., auditor–client relationships, personal and behavioral characteristics, internal control overrides), most recent studies recognized that this strategy is problematic for most stakeholders because internal data sources are not readily available to investors, auditors, or regulators (Albrecht, Albrecht, and Albrecht 2008; Cecchini et al. 2010; Dechow et al. 2011). Accordingly, we limit our discussion of prior financial fraud detection research to studies that have used publicly available (i.e., external) information. These studies have generally used measures derived from public financial statements in combination with statistical and/or machine learning-based classification methods. A summary of related fraud detection research using external data is presented in Table 2, which shows the author and the year of publication, a brief description of the feature set, the method used, the data set, and the resulting overall classification accuracy and fraud detection rate. From Table 2, we see that the time line of prior studies spans a 15-year period from Persons (1995) to Dikmen and Küçükkocaoğlu (2010), which underscores the continuing importance of financial fraud detection to both academics and practitioners.

With respect to the feature sets utilized, prior studies unvaryingly employed data taken from annual financial reports. Most studies used 8 to 10 financial measures (e.g., Beneish 1999a; Green and Choi 1997; Kirkos et al. 2007; Persons

Table 2. Prior Financial Fraud Detection Studies Using Financial Statement Data

Study	Annual Statement-based Feature Set	Classification Method(s)	Data Set	Results
Persons (1995)	10 financial measures from previous year	Logistic regression	200 firm-years; 100 fraud, 100 non-fraud	Overall: 71.5%; Fraud: 64.0%
Green and Choi (1997)	5 financial and 3 accounting measures	Neural net	95 firm-years; 46 fraud, 49 non-fraud	Overall: 71.7%; Fraud: 68.4%
Fanning and Cogger (1998)	26 financial and 36 accounting measures	Discriminant analysis, Logistic regression, Neural net	204 firm-years; 102 fraud, 102 non-fraud	Overall: 63.0%; Fraud: 66.0%
Summers and Sweeney (1998)	6 financial measures	Logistic regression	102 firm-years; 51 fraud, 51 non-fraud	Overall: 59.8%; Fraud: 67.8%
Beneish (1999a)	8 financial measures	Probit regression	2,406 firm-years; 74 fraud, 2,332 non-fraud	Overall: 89.5%; Fraud: 54.2%
Spathis (2002)[a]	10 financial measures	Logistic regression	76 firm-years; 38 fraud, 38 non-fraud	Overall: 84.2%; Fraud: 84.2%
Spathis et al. (2002)[a]	10 financial measures	Logistic regression, UTADIS	76 firm-years; 38 fraud, 38 non-fraud	Overall: 75.4%; Fraud: 64.3%
Lin et al. (2003)	6 financial and 2 accounting measures	Logistic regression, Neural net	200 firm-years; 40 fraud, 160 non-fraud	Overall: 76.0%; Fraud: 35.0%
Kaminski et al. (2004)	21 financial measures	Discriminant analysis	158 firm-years; 79 fraud, 79 non-fraud	Overall: 53.8%; Fraud: 21.7%
Kirkos et al. (2007)[a]	10 financial measures	Bayesian net, ID3 decision tree, Neural net	76 firm-years; 38 fraud, 38 non-fraud	Overall: 90.3%; Fraud: 91.7%
Gaganis (2009)[a]	7 financial measures	Discriminant analysis, Logistic regression Nearest neighbor, Neural net, SVM, UTADIS	398 firm-years; 199 fraud, 199 non-fraud	Overall: 87.2%; Fraud: 87.8%
Cecchini et al. (2010)	23 financial variables used to generate ratios	SVM using custom financial kernel	3,319 firm-years; 132 fraud, 3,187 non-fraud	Overall: 90.4% Fraud: 80.0%
Dikmen and Küçükkocaoğlu (2010)[b]	10 financial measures	Three-phase cutting plane algorithm	126 firm-years; 17 fraud, 109 non-fraud	Overall: 67.0%; Fraud: 81.3%
Dechow et al. (2011)	7 financial measures	Logistic regression	79,651 firm-years; 293 fraud 79,358 non-fraud	Overall: 63.7%; Fraud: 68.6%

[a]Data taken from Greek firms. [b]Data taken from Turkish firms.

1995; Spathis 2002; Summers and Sweeney 1998). While most prior studies using larger feature sets did not attain good results (e.g., Fanning and Cogger 1997; Kaminski et al. 2004), Cecchini et al. (2010) had greater success using 23 seed financial variables to automatically generate a large set of financial ratios.

The most commonly used classification methods were logistic regression, neural networks, and discriminant analysis, while decision trees, Bayesian networks, and support vector machines (SVM) have been applied in more recent studies (Cecchini et al. 2010; Gaganis 2009; Kirkos et al. 2007). The number of fraud firms in the data sets ranged from 38 firms to 293 firms. Most studies used a pair-wise approach in which the number of non-fraud firms was matched with the number of fraud firms (Fanning and Cogger 1998; Gaganis 2009; Kirkos et al. 2007; Persons 1995; Spathis 2002; Summers and Sweeney 1998). However, a few studies had significantly larger sets of non-fraud firms (Beneish 1999a; Cecchini et al. 2010; Dechow et al. 2011).

In terms of results, the best performance values were achieved by Cecchini et al. (2010), Gaganis (2009), Kirkos et al. (2007), and Spathis (2002). Only these four studies had overall accuracies and fraud detection rates of more than 80 percent. The latter three were all conducted using a data set composed of Greek firms (mostly in the manufacturing sector), which were governed by Greek legislation and Athens Stock Exchange regulations regarding what is classified as unusual behavior by a firm. Given the differences in auditing and reporting standards between Greece and the United States, as well as the larger international community (Barth et al. 2008), it is unclear how well those methods generalize/ translate to other settings. Cecchini et al. used an SVM classifier that incorporated a custom financial kernel. The financial kernel was a graph kernel that used input financial variables to implicitly derive numerous financial ratios. Their approach attained a fraud detection rate of 80 percent on a 25 fraud-firm test set.

With respect to the remaining studies, Beneish (1999a) attained an overall accuracy of 89.5 percent, but this was primarily due to good performance on non-fraud firms, whereas the fraud detection rate was 54.2 percent. With the exception of Cecchini et al. no other prior study on U.S. firms has attained a fraud detection rate of more than 70 percent.

Our analysis of these studies motivated several refinements that are incorporated in our meta-learning framework, namely (1) the inclusion of organizational and industry-level context information, (2) the utilization of quarterly and annual statement-based data, and (3) the adoption of more robust fraud classification methods. The MetaFraud framework and its components are discussed in detail in the following sections.

Using Meta-Learning as a Kernel Theory for Financial Fraud Detection

As mentioned in the previous section, and evidenced by Table 2, prior studies using data from U.S. firms have generally attained inadequate results, with fraud detection rates typically less than 70 percent. These results have caused some to suggest that data based on financial statements is incapable of accurately identifying financial fraud (at least in the context of U.S. firms). In one of the more recent studies on U.S. firms, Kaminski et al. (2004, p. 17) attained results only slightly better than chance, causing the authors to state, "These results provide empirical evidence of the limited ability of financial ratios to detect fraudulent financial reporting." A more recent study conducted by researchers at Pricewaterhouse Coopers attained fraud detection rates of 64 percent or lower (Bay et al. 2006). The limited performance of these and other previous studies suggests that the financial measures and classification methods employed were insufficient. Prior studies have generally relied on 8 to 10 financial ratios, coupled with classifiers such as logistic regression or neural networks. In light of these deficiencies, more robust approaches for detecting financial fraud are needed (Cecchini et al. 2010).

Design science is a robust paradigm that provides concrete prescriptions for the development of IT artifacts, including constructs, models, methods, and instantiations (March and Smith 1995). In the design science paradigm, "Methods define processes. They provide guidance on how to solve problems, that is, how to search the solution space" (Hevner et al. 2004, p. 79). Several prior studies have utilized a design science approach to develop BI technologies encompassing methods and instantiations (Abbasi and Chen 2008a; Chung et al. 2005). Accordingly, we were motivated to develop a framework for enhanced financial fraud detection (i.e., a method).

When creating IT artifacts in the absence of sufficient design guidelines, many studies have emphasized the need for design theories to help govern the development process (Abbasi and Chen 2008a; Markus et al. 2002; Storey et al. 2008; Walls et al. 1992). We used meta-learning as a kernel theory to guide the development of the proposed financial fraud detection framework (Brazdil et al. 2008). In the remainder of this section, we present an overview of meta-learning and discuss how meta-learning concepts can be utilized to address the aforementioned research gaps, resulting in enhanced financial fraud detection capabilities. Testable research hypotheses are also presented. A framework based on meta-learning for financial fraud detection is then described and evaluated.

Meta-Learning

Meta-learning is a specialized form of machine learning that uses the expertise acquired through machine learning or data mining processes to increase the quality of results obtained in future applications (Brazdil et al. 2008). While *machine learning* provides a multitude of algorithms to complete a task without offering guidance about which particular algorithms to use in a given context, in contrast, *meta-learning* provides a way to learn about the learning process itself to obtain knowledge about which underlying features and algorithms can be most efficiently applied (Brazdil et al. 2008). We posit that a meta-learning approach is especially appropriate for financial fraud detection because of the complex, dynamic, and adversarial nature of this problem domain (Virdhagriswaran and Dakin 2006).

Meta-learning was developed in the late 1980s and early 1990s by a number of researchers who sought to integrate several machine learning strategies to enhance overall accuracy (e.g., Wolpert 1992). The term *meta-learning* was coined by Chan and Stolfo (1993), who proposed a method for combining the outputs of multiple machine-learning techniques in a self-adaptive way to improve accuracy. The method has since evolved into several active streams of research in a variety of application domains (Brazdil et al. 2008; Vilalta and Drissi 2002). Collectively, meta-learning provides an array of prescriptions for improving machine learning capabilities pertaining to a particular problem task.

In machine learning, learning *bias* refers to any preference for choosing one hypothesis that explains the data over other equally acceptable hypotheses (Mitchell 1997). Meta-learning can be defined as the ability to learn from experience when different biases are appropriate for a particular problem (Brazdil et al. 2008; Rendell et al. 1987). Meta-learning, therefore, differs from conventional or base-learning techniques in that it enriches the model hypothesis space, which increases the likelihood of finding good models. Yet, the space itself is of fixed form: no dynamic selection of bias takes place. A critical need in the BI area is to devise methods of bias selection that can adapt to changes that may occur in the problem domain (Vilalta and Drissi 2000).

At another level, two key facets of meta-learning are declarative and procedural bias (Brazdil et al. 2008). Declarative bias specifies the representation of the space of hypotheses; it is governed by the quantity and type of attributes incorporated (i.e., the feature space). Procedural bias pertains to the manner in which classifiers impose constraints on the ordering of the inductive hypotheses.

An effective meta-learning strategy dynamically identifies appropriate levels of declarative and procedural bias for a given classification task (Vilalta and Drissi 2002). Declarative bias can be manipulated by altering the feature space (i.e., via expansion or contraction), while procedural bias can be improved by selecting a suitable predictive model or combination of models (Giraud-Carrier et al. 2004; Vilalta and Drissi 2002). In the remainder of this section, we describe how declarative bias can be improved by supplementing existing measures based on annual statements with context-based features and quarterly data. We also discuss how procedural bias can be enhanced by using stacked generalization and adaptive learning.

Improving Declarative Bias by Incorporating Context-Based Features and Quarterly Data

Prior financial fraud detection studies have typically utilized feature sets composed of fewer than 30 financial measures (with most using fewer than 10). Moreover, prior studies have relied on measures derived exclusively from annual statements. Consequently, the feature sets incorporated lacked representational richness and were simply not large enough to generate appropriate hypothesis spaces for the classification methods utilized. Meta-learning approaches for expanding the feature space in order to improve declarative bias include feature construction (i.e., the use of seed features to generate additional features) as well as the extraction of new and existing features from additional (parallel) data sets (Brazdil et al. 2008; Vilalta et al. 2004). Both of these approaches are discussed below.

Industry-Level and Organizational Context-Based Features

Financial fraud detection relies on identification of financial irregularities. Prior studies have used measures derived from firms’ financial statements (Kaminski et al. 2004; Lin et al. 2003; Summers and Sweeney 1998). While there is strong theoretical evidence justifying the use of financial measures (Beneish 1999a; Dikmen and Küçükkocaoğlu 2010), the manner in which they have been utilized merely provides a non-holistic snapshot, without appropriate consideration of the context surrounding these measures. Two important types of contextual information generally omitted by previous studies are organizational and industry-level contexts.

Organizational context information can be derived by comparing a firm's financial performance relative to its performance in prior periods. Auditors commonly compare firms' financial measures across consecutive time periods in order to identify potential irregularities (Ameen and Strawser 1994; Green and Choi 1997). Further, prior financial fraud detection studies suggest that utilizing measures from the preceding year can provide useful information (Cecchini et al. 2010; Fanning and Cogger 1998; Persons 1995; Virdhagriswaran and Dakin 2006). Consideration of data across multiple time periods can reveal organizational trends and anomalies that are often more insightful than information derived from single-period snapshots (Chen and Du 2009; Coderre 1999; Green and Calderon 1995; Kinney 1987).

Industry-level context information can be derived by comparing a firm's financial performance relative to the performance of its industry peers. Prior studies have found that certain industries have financial statement irregularity patterns that are unique and distinctly different from other industries (Maletta and Wright 1996). Beasley et al. (2000) analyzed financial fraud cases arising in three industries over a two-decade period. They found that technology companies and financial-service firms differed considerably in terms of their values for accounting measures as well as the categories of fraud that were pervasive in the two industries. These findings suggest that industry-level context information could be highly useful for automated financial fraud detection.

Despite the potential of industry-level context information to aid in fraud detection, thus far this information has not been utilized in fraud detection research. Fanning and Cogger (1998) noted that industry-level context information represented an important and unexplored future research direction: "Certain variables may become useful classifiers when examined in a sample stratified by industry" (p. 37). Similarly, Spathis (2002, p. 218) claimed that "Industry standing probably would provide additional valuable information." In this study we extend and exploit relevant contextual information.

The use of financial measures without appropriate contextual information results in parsimonious input feature spaces. Accordingly, we posit that feature sets utilizing yearly or quarterly financial measures in conjunction with organizational and industry-level context information will result in improved financial fraud detection performance in terms of overall accuracy and class-level f-measure, precision, and recall.

H1a: *Combining yearly financial measures with organizational context features will outperform the use of yearly financial measures alone.*

H1b: *Combining yearly financial measures with industry-level context features will outperform the use of yearly financial measures alone.*

H1c: *Combining yearly financial measures with industry-level and organizational context features will outperform the use of yearly financial measures alone.*

H1d: *Combining quarterly financial measures with organizational context features will outperform the use of quarterly financial measures alone.*

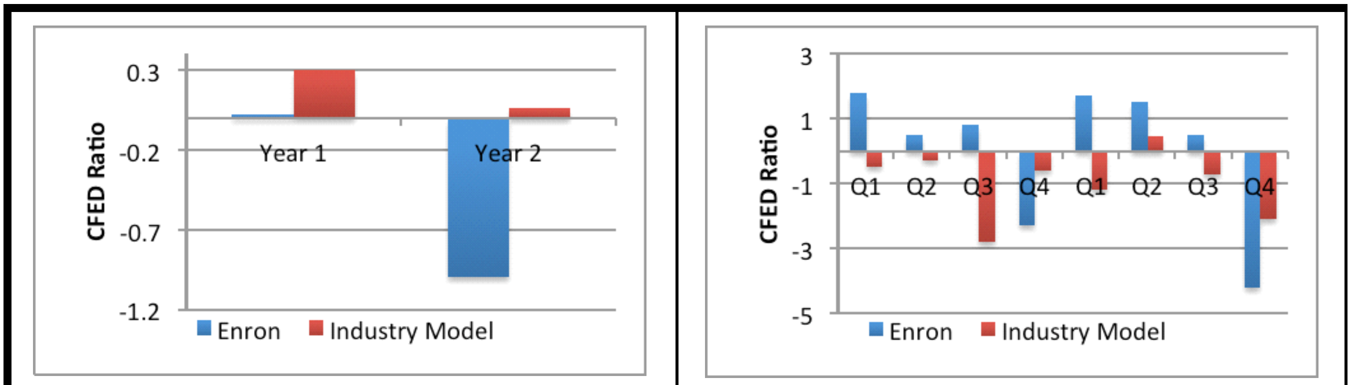
H1e: *Combining quarterly financial measures with industry-level context features will outperform the use of quarterly financial measures alone.*

H1f: *Combining quarterly financial measures with industry-level and organizational context features will outperform the use of quarterly financial measures alone.*

Quarterly Statement-Based Features

Most prior studies of financial fraud used only information based on annual statements (Yue et al. 2007). However, using both quarterly and annual data can provide a more robust feature space for financial fraud detection (Albrecht, Albrecht, and Albrecht 2004). On the practice side, Putra provided an extended argument for including an examination of both quarterly and annual data in assessments of financial statement fraud (Accounting Financial and Tax 2009). This has been affirmed by several recent research studies. Dechow et al. (2011) proposed an F-score designed to detect accounting fraud from misrepresentations made on quarterly or annual financial statements. Dull and Tegarden (2004) included quarterly financial data in their development of a control-chart approach to monitoring key financial processes. Chai et al. (2006) used quarterly financial statements as the basis for developing rankings of the degree to which firms match rules associated with financial fraud.

Albrecht, Albrecht, and Albrecht (2004) argued that annual numbers are often too aggregated to reveal significant differences, and that concealment methods differ dramatically between the end of the first three quarters and the end of the year-end quarter. In their analysis, one large company that manipulated its financial statements used unsupported topside



(a) Yearly **(b) Quarterly**

Figure 1. Yearly and Quarterly Cash Flow Earnings Differences (CFED) Ratios for the Enron Fraud

entries—entries that remove the discrepancy between actual operating results and published financial reports—as the major way to commit fraud at the end of the first three quarters, but harder-to-detect, sophisticated revenue and expense frauds at the end of the year. Another company used topside entries at the end of the first three quarters but shifted losses and debt to unconsolidated, related entities at the end of the year.

As a specific example, consider the cash flow earnings difference ratio for the Enron fraud (Figure 1). Figure 1a shows this ratio for Enron and its industry model, over a two-year period, using only annual numbers. While Enron's values are slightly lower than the industry model's, the graph exhibits no recognizable pattern. Figure 1b shows this ratio for Enron and its industry model on a quarterly basis. Enron's figures are primarily positive for the first three quarters, and then sharply negative in the fourth quarter. Throughout the first three quarters, Enron's management was using various accounting manipulations to make their income statement look better (Albrecht, Albrecht, and Albrecht 2004; Kuhn and Sutton 2006). At the end of the year, Enron's management "corrected" the discrepancy by shifting those losses to off-balance sheet, nonconsolidated special purpose entities (now called variable interest entities). This difference in manipulation methods between quarters is not apparent when analyzing annual data, but it shows up in the cash flow earnings difference ratio when used with quarterly data.

The above evidence supports the argument that incorporating different levels of information granularity is important in detecting financial fraud. Accordingly, we believe that using context-based features taken from both annual and quarterly statements will improve financial fraud detection performance over using features from either category alone.

H2a: *Combining yearly and quarterly statement-based features will outperform the use of only yearly features in terms of fraud detection performance.*

H2b: *Combining yearly and quarterly statement-based features will outperform the use of only quarterly features in terms of fraud detection performance.*

Improving Procedural Bias Using Stacked Generalization and Adaptive Learning

Prior financial fraud detection studies have used several different classification methods, with logistic regression, neural networks, and discriminant analysis being the most common. However, no single classifier has emerged as a state-of-the-art technique for detecting financial fraud (Fanning and Cogger 1998; Gaganis 2009; Kirkos et al. 2007). Therefore, the need remains for enhanced classification approaches capable of improving procedural bias. Meta-learning strategies for enhancing procedural bias include stacked generalization and adaptive learning (Brazdil et al. 2008). Stacked generalization involves the use of a top-level learner capable of effectively combining information from multiple base learners (Vilalta and Drissi 2002; Wolpert 1992). Adaptive learning entails constant relearning and adaptation (i.e., dynamic bias selection) to changes in the problem environment, including concept drift (Brazdil et al. 2008).

Stacked Generalization

Classifiers are often diverse with respect to their categorization performance, patterns, and biases (Kuncheva and Whitaker 2003; Lynam and Cormack 2006). In addition, classifiers frequently provide complementary information that could be useful if exploited in unison (Tsoumakas et al. 2005). Prior financial fraud detection studies utilizing multiple classification methods have also observed some levels of noncorrelation between the classifiers' predictions, even when overall accuracies were equivalent. For instance, Fanning and Cogger (1998) attained the best overall accuracy using a neural network, yet the fraud detection rates were considerably higher (i.e., 12 percent) when using discriminant analysis. While comparing logistic regression and neural network classifiers, Lin et al. (2003) noted that the two classifiers achieved somewhat comparable overall accuracies; however, logistic regression had 11 percent better performance on non-fraudulent firms, while the neural network obtained 30 percent higher fraud detection rates. Similarly, Gaganis (2009) observed equally good overall results using a UTADIS scoring method and a neural network; however, the respective false positive and false negative rates for the two methods were exact transpositions of one another. These findings suggest that methods capable of conjunctively leveraging the strengths of divergent classifiers could yield improved financial fraud detection performance.

Stacked generalization (also referred to as *stacking*) provides a mechanism for harnessing the collective discriminatory power of an ensemble of heterogeneous classification methods (Wolpert 1992). Stacking involves the use of a top-level classification model capable of learning from the predictions (and classification biases) of base-level models in order to achieve greater classification power (Brazdil et al. 2008; Hansen and Nelson 2002; Ting and Witten 1997; Wolpert 1992). As Sigletos et al. (2005, p. 1751) noted, "The success of stacking arises from its ability to exploit the diversity in the predictions of base-level classifiers and thus predicting with higher accuracy at the meta-level." This ability to learn from underlying classifiers makes stacking more effective than individual classifier-based approaches or alternate fusion strategies that typically combine base-level classifications using a simple scoring or voting scheme (Abbasi and Chen 2009; Dzeroski et al. 2004; Hu and Tsoukalas 2003; Lynam and Cormack 2006; Sigletos et al. 2005). Consequently, stacking has been effectively utilized in related studies on insurance and credit card fraud detection, outperforming the use of individual classifiers (Chan et al. 1999; Phua et al. 2004).

Given the performance diversity associated with fraud detection classifiers employed in prior research, the use of stacked generalization is expected to be highly beneficial, facilitating enhanced financial fraud detection capabilities over those achieved by individual classifiers. We predict this performance gain will be actualized irrespective of the specific context-based feature set utilized.

H3a: *When yearly context-based features are used, stack classifiers will outperform individual classifiers in terms of fraud detection performance.*

H3b: *When quarterly context-based features are used, stack classifiers will outperform individual classifiers in terms of fraud detection performance.*

Adaptive Learning

While fraud lies behind many of the largest bankruptcies in history, there are considerable differences between the types of frauds committed and the specific obfuscation tactics employed by previous firms. For instance, the $104 billion WorldCom fraud utilized a fairly straightforward expense capitalization scheme (Zekany et al. 2004). In contrast, the $65 billion Enron fraud was highly complex and quite unique; the use of special-purpose entities as well as various other tactics made detection very difficult (Kuhn and Sutton 2006). These examples illustrate how financial fraud cases can be strikingly different in terms of their complexity and nature. Effective financial fraud detection requires methods capable of discovering fraud across a wide variety of industries, reporting styles, and fraud types over time.

Fraud detection is a complex, dynamic, and evolving problem (Abbasi et al. 2010; Bolton and Hand 2002). Given the adversarial nature of fraud detection, the classification mechanisms used need constant revision (Abbasi et al. 2010). Adaptive learning methods have the benefit of being able to relearn, in either a supervised or semi-supervised capacity, as new examples become available (Brazdil et al. 2008; Fawcett and Provost 1997). The ability to adaptively learn is especially useful in the context of financial fraud because the risk environment surrounding fraud is expected to change, making it more difficult to detect (Deloitte 2010). Virdhagriswaran and Dakin (2006, p. 947) noted that adaptive learning could greatly improve fraud detection capabilities by "identifying compensatory behavior" by fraudsters "trying to camouflage their activities." Similarly, Fawcett and Provost (1997, p. 5) observed that "it is important that a fraud detection system adapt easily to new conditions. It should be able to notice new patterns of fraud." An adaptive, learning-based classifier that is aware of its changing environment and able to constantly retrain itself accordingly, should outperform its static counterpart.

H4: The use of an adaptive learning mechanism, capable of relearning as new information becomes available, will outperform its static counterpart in terms of fraud detection performance.

Collective Impact Attributable to Improving Declarative and Procedural Bias

Based on the previous four hypotheses, we surmise that financial fraud detection methods incorporating meta-learning principles pertaining to the improvement of declarative and procedural bias are likely to provide enhanced discriminatory potential (Brazdil et al. 2008). Specifically, we expect that the use of industry and organizational context information derived from both yearly and quarterly statements for declarative bias improvement (H1–H2), coupled with stacked generalization and adaptive learning for procedural bias improvement (H3–H4), will facilitate improvements in overall financial fraud detection capabilities. Accordingly, we hypothesized that, collectively, a meta-learning framework that incorporates these principles will outperform existing state-of-the art financial fraud detection methods.

H5: A meta-learning framework that includes appropriate provisions for improving declarative and procedural bias in concert will outperform existing methods in terms of fraud detection performance.

Prior studies have effectively used ensemble approaches in concert with semi-supervised learning (Balcan et al. 2005; Ando and Zhang 2007; Zhou and Goldman 2004). For instance, Zhou and Li (2005) markedly improved the performance of underlying classifiers on several test beds, in various application domains, by using a three-classifier ensemble in a semi-supervised manner. It is, therefore, conceivable that such ensemble-based semi-supervised methods could also facilitate improved procedural bias for financial fraud detection. However, given the reliance of such methods on voting schemes across base classifiers (Balcan et al. 2005; Zhou and Li 2005), we believe that ensemble semi-supervised learning methods will underperform meta-learning strategies that harness the discriminatory potential of stacked generalization and adaptive learning.

H6: A meta-learning framework that includes stacked generalization and adaptive learning will provide improved procedural bias over existing ensemble-based semi-supervised learning methods, resulting in enhanced financial fraud detection performance.

A Meta-Learning Framework for Financial Fraud Detection

We propose a meta-learning framework for detecting financial fraud to address the research gaps and related hypotheses presented in the previous section, namely (1) the use of organizational and industry contextual information, (2) the use of quarterly and annual data, and the use of more robust classification methods using (3) stacked generalization and (4) adaptive learning. In this section, we demonstrate how our meta-learning framework fulfills each of these requirements to enhance financial fraud detection.

The MetaFraud framework utilizes a rich feature set, numerous classification methods at the base and stack level, and an adaptive learning algorithm. Each component of the framework (shown in Figure 2) is intended to enhance financial fraud detection capabilities. Beginning with a set of yearly and quarterly seed ratios, industry-level and organizational context-based features are derived to create the yearly and quarterly feature sets (bottom of Figure 2). These feature sets are intended to improve declarative bias. The features are used as inputs for the yearly and quarterly context-based classifiers. The classifications from these two categories of classifiers are then used as input for a series of stack classifiers. The adaptive, semi-supervised learning algorithm, shown at the top of Figure 2, uses the stack classifiers' predictions to iteratively improve classification performance. The stack classifiers and adaptive learning algorithm are intended to improve procedural bias.

As a simple example, think of Stack_Classifier1 as an SVM, which takes input from bottom (1) yearly context-based classifiers and (2) quarterly context-based classifiers; such as SVM, J48, BayesNet, NaiveBayes etc. Stack_Classifier2 might be a J48 classifier, which accepts inputs from the same bottom yearly and quarterly context-based classifiers: SVM, J48, BayesNet, NaiveBayes etc. Output from the stack classifiers is aggregated and input to the adaptive learner. The four components of the framework are closely related to the research hypotheses. Each of these components is explained below.

Financial Fraud Detection Feature Sets

The framework uses two feature sets based on yearly and quarterly information, respectively. The yearly feature set uses 12 seed financial ratios derived from annual statements to generate additional organizational and industry-level context features. The quarterly feature set uses the quarterly versions of the same 12 ratios, resulting in 48 seed measures, to generate additional quarterly organizational and industry-level context features. In the ensuing sections, we describe the financial ratios utilized followed by a discussion of the yearly and quarterly feature sets.

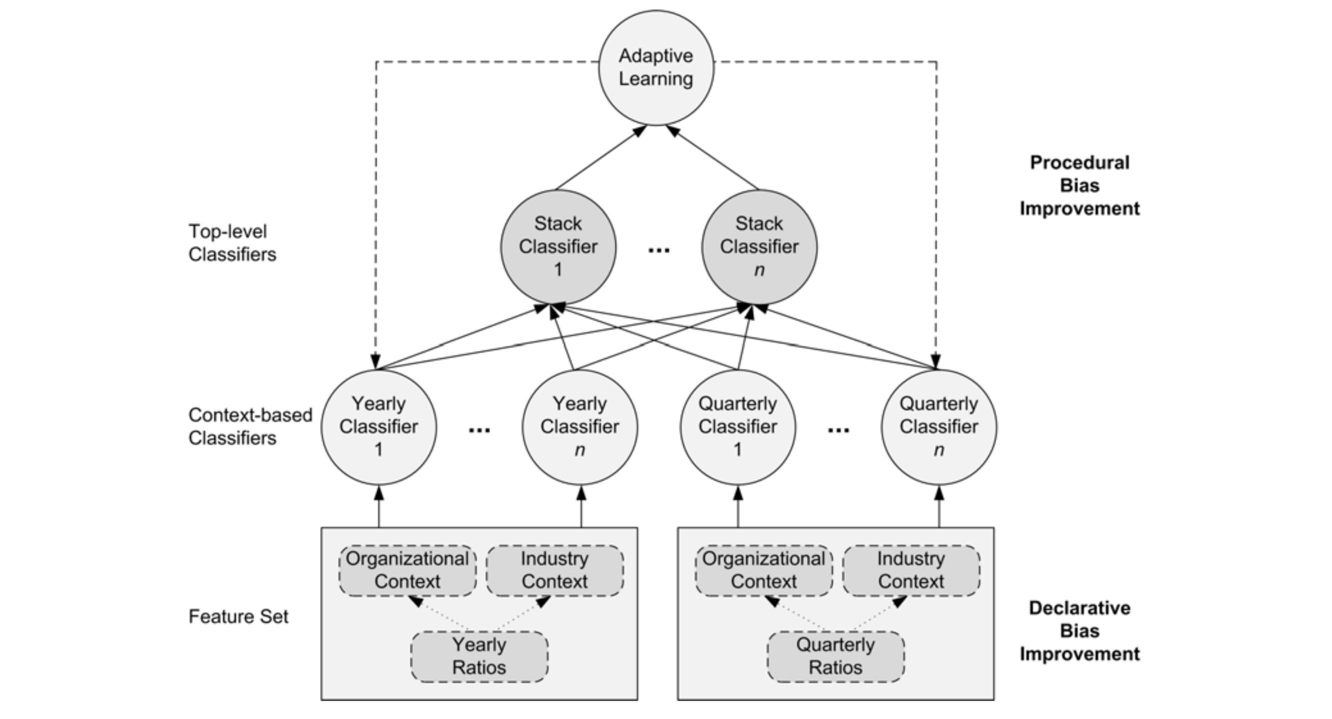


Figure 2. The MetaFraud Framework

Financial Ratios

Our selection of the seed financial ratios was influenced by prior financial fraud detection studies. These 12 ratios are described below.

1. *Asset Quality Index (AQI)*: *AQI* is the ratio of non-current assets other than property, plant, and equipment, to total assets, for time period *t* relative to time period *t-1*. An *AQI* greater than 1 indicates that the firm has potentially increased its involvement in cost deferral, a possible indicator of asset overstatement fraud (Beneish 1999a; Dikman and Küçükkocaoğlu 2010).

2. *Asset Turnover (AT)*: *AT* is the ratio of net sales to total assets. When revenue fraud is being committed, net sales are often increased artificially and rapidly, resulting in a large *AT* value (Cecchini et al. 2010; Kirkos et al. 2007; Spathis 2002; Spathis et al. 2002).

3. *Cash Flow Earnings Difference (CFED)*: *CFED* assesses the impact of accruals on financial statements (Beneish 1999a; Dechow et al. 2011). This ratio is often positive when revenue fraud is occurring or when employees are engaging in cash theft.

4. *Days Sales in Receivables (DSIR)*: *DSIR* is the ratio of day sales in receivables in period *t* to the corresponding measure in period *t-1*. Firms engaging in revenue fraud often add fictitious receivables, causing *DSIR* to increase (Green and Choi 1997; Kaminski 2004; Lin 2003).

5. *Depreciation Index (DEPI)*: *DEPI* is the ratio of the rate of depreciation in period *t-1* as compared to period *t*. Fictitious assets accelerate the depreciation rate, resulting in smaller values for *DEPI* (Beneish 1999a; Cecchini et al. 2010; Dikman and Küçükkocaoğlu 2010).

6. *Gross Margin Index (GMI)*: *GMI* is the ratio of the gross margin in period *t-1* to the gross margin in period *t*. A *GMI* greater than 1 suggests that gross margins have deteriorated, a condition rarely encountered when a firm is engaging in revenue fraud (Beneish 1999a; Lin 2003).

7. *Inventory Growth (IG)*: *IG* assesses whether inventory has grown in period *t* as compared to period *t-1*. *IG* is used to detect whether ending inventory is being overstated to decrease cost of goods sold and increase gross margin (Cecchini et al. 2010; Dikman, and Küçükkocaoğlu 2010; Persons 1995).

8. *Leverage (LEV)*: *LEV* is the ratio of total debt to total assets in period *t* relative to period *t-1*. *LEV* is used to detect whether firms are fictitiously including assets on the balance sheet without any corresponding debt (Cecchini et al. 2010; Beneish 1999a; Kirkos, et al 2007; Persons 1995; Spathis 2002; Spathis et al. 2002).

9. *Operating Performance Margin (OPM)*: *OPM* is calculated by dividing net income by net sales. When fraudulent firms add fictitious sales revenues, they often fall to the bottom line without additional costs, thus inflating the value of *OPM* (Cecchini et al. 2010; Persons 1995; Spathis 2002; Spathis et al. 2002).

10. *Receivables Growth (RG)*: *RG* is the amount of receivables in period *t* divided by the amount in period *t-1*. Firms engaging in revenue fraud often add fictitious revenues and receivables, thereby increasing *RG* (Cecchini et al. 2010; Dechow et al. 2011; Summers and Sweeney 1998).

11. *Sales Growth (SG)*: *SG* is equal to net sales in period *t* divided by net sales in period *t-1*. In the presence of revenue fraud, the value of SG generally increases (Beneish 1999a; Cecchini et al. 2010; Gaganis 2009; Dikman and Küçükkocaoğlu 2010; Persons 1995; Summers and Sweeney 1998).

12. *SGE Expense (SGEE)*: *SGEE* is calculated by dividing the ratio of selling and general administrative expenses to net sales in period *t* by the same ratio in period *t-1*. When firms are engaging in revenue fraud, SGE expenses represent a smaller percentage of the artificially inflated net sales, thereby causing *SGEE* to decrease (Beneish 1999a; Cecchini et al. 2010; Dikman and Küçükkocaoğlu 2010).

Yearly and Quarterly Context-Based Feature Sets

The yearly and quarterly context-based feature sets used the aforementioned seed ratios to derive industry-level and organizational context features. The context features were developed using feature construction: the process of applying constructive operators to a set of existing features in order to generate new features (Matheus and Rendell 1989). Feature construction facilitates the fusion of data and domain knowledge to construct features with enhanced discriminatory potential (Dybowski et al. 2003). In meta-learning, feature construction is recommended as a mechanism for improving declarative bias in situations where the hypothesized set generated by a particular feature set needs to be expanded (Brazdil et al. 2008). In prior BI studies, feature construction was used to derive complex and intuitive financial ratios and metrics from (simpler) seed accounting variables (Piramuthu et al. 1998; Zhao et al. 2009). These new features were often generated by combining multiple seed measures using arithmetic operators such as multiplication and division (Langley et al. 1986; Zhao et al. 2009).

We used subtraction and division operators to construct new features indicative of a firm's position relative to its own prior performance (organizational context) or its industry (industry-level context). First, the organizational context features were constructed by computing the difference between (–) and the ratio of (/) the firms' seed financial ratios (described in the previous section) in the current time period relative to their values for the same ratios in the previous time period.

Second, to generate the industry-level context features, we developed industry-representative models designed to characterize what is normal for each industry. Each firm's industry affiliation was defined by its North American Industry Classification System (NAICS) code. NAICS was used since it is now the primary way Compustat and other standards bodies reference industries. Two types of models were developed. Top-5 models were created by averaging the data from the five largest companies in each industry-year (in terms of sales), and then generating the 12 seed financial ratios from these averaged values. Hence, each industry had a single corresponding top-5 model. Closest-5 models were created for each firm by averaging the data from the five companies from the same industry-year that were most similar in terms of sales. Hence, each firm had a corresponding closest-5 model. The intuition behind using these two types of models was that the top-5 models represent the industry members with the greatest market share (and therefore provide a single reference model for all firms in the industry), while closest-5 models represent the firms' industry peers. As revealed in the evaluation section, both types of models had a positive impact on fraud detection performance.

For a given model, total assets were calculated as the average of total assets of the five companies, while the accounts receivable was the average accounts receivable of the same companies. Multiple firms were used to construct each model in order to smooth out any non-industry-related fluctuations attributable to individual firms (Albrecht, Albrecht, and Dunn 2001). On the other extreme, using too many firms produced models that were too aggregated. Therefore, in our preliminary analysis, we explored the use of different numbers of firms and found that using five provided the best balance. The industry-level context features were then constructed by computing the difference between (–) and the ratio of (/) the

Table 3. Yearly Context-Based Feature Set

Type	Description	Quantity
Yearly financial ratios	R1, R2,…R12	12
Industry-level context: Top-5 Model	R1-T1, R2-T2,…R12-T12	12
	R1/T1, R2/T2,…R12/T12	12
Industry-level context: Closest-5 Model	R1-C1, R2-C2,…R12-C12	12
	R1/C1, R2/C2,…R12/C12	12
Organizational context	R1-P1, R2-P2,…R12-P12	12
	R1/P1, R2/P2,…R12/P12	12
Total		**84**

Table 4. Quarterly Context-Based Feature Set

Type	Description	Quantity
Quarterly financial ratios	R1Q1, R2Q1,…R12Q4	48
Industry-level context: Top-5 Model	R1Q1-T1Q1, R2Q1-T2Q1,…R12Q4-T12Q4	48
	R1Q1/T1Q1, R2Q1/T2Q1,…R12Q4/T12Q4	48
Industry-level context: Closest-5 Model	R1Q1-C1Q1, R2Q1-C2Q1,…R12Q4-C12Q4	48
	R1Q1/C1Q1, R2Q1/C2Q1,…R12Q4/C12Q4	48
Organizational context	R1Q2-R1Q1, R1Q3-R1Q2,…R12Q4-R12Q3	48
	R1Q2/R1Q1, R1Q3/R1Q2,…R12Q4/R12Q3	48
Total		**336**

firms' seed financial ratios and those of their respective top-5 and closest-5 industry models.

Table 3 shows the yearly context-based feature set. For each firm, we used the 12 seed year-level ratios (R1–R12) as well as 48 industry-level context features derived using the firms' corresponding top-5 (T1–T12) and closest-5 (C1–C12) industry models. For instance, the industry-level context feature R1-C1 signifies the difference between a firm's *Asset Quality Index* and that of its closest-5 industry model. An additional 24 organizational context features were constructed using the firms' ratios from the previous year (P1–P12). This resulted in a feature set composed of 84 attributes.

Table 4 shows the quarterly context-based feature set. Each of the 12 seed financial ratios was derived from all four quarterly statements, resulting in 48 core features (R1Q1–R12Q4). These were used to generate 96 top-5 model-based industry-level context features (e.g., R1Q1–T1Q1 and R1Q1/T1Q1), and 96 closest-5 model-based features (e.g., R1Q1–C1Q1 and R1Q1/C1Q1). Furthermore, 96 organizational context features were constructed by comparing the firms' ratios in a particular quarter against those from the previous quarter (e.g., R1Q2/R1Q1 denotes the ratio of a firm's *Asset Quality Index* in quarter 2 as compared to quarter 1). This resulted in a quarterly feature set composed of 336 attributes.

Yearly and Quarterly Context-Based Classifiers

The yearly and quarterly context-based feature sets were coupled with an array of supervised learning classification methods. Prior studies have mostly used logistic regression and neural network classifiers (Fanning and Cogger 1998; Green and Choi 1997; Lin et al. 2003; Persons 1995; Spathis 2002). However, additional classification methods have also attained good results for financial fraud detection (e.g., Kirkos et al. 2007), as well as related fraud detection problems (e.g., Abbasi and Chen 2008b; Abbasi et al. 2010), including support vector machines, tree classifiers, and Bayesian methods. Given the lack of consensus on best methods, as described earlier, a large number of classifiers were used in order to improve overall fraud detection performance. Moreover, the use of a large set of classifiers also provided a highly useful confidence-level measure, which is described in the evaluation section.

Accordingly, we incorporated several classifiers in addition to logistic regression and neural networks. Three support vector machines (SVM) classifiers were utilized: linear, polynomial, and radial basis function (RBF) kernels (Vapnik 1999). Two Bayesian classifiers were used: Naïve Bayes and Bayesian Networks (Bayes 1958). Various tree-based classifiers were employed, including the J48 decision tree, Naïve Bayes Tree (NBTree), ADTree, Random Forest, and REPTree (Breiman 2001; Freund and Mason 1999; Kohavi 1996; Quinlan 1986). Two rule-based classifiers were also included: nearest neighbor (NNge) and JRip (Cohen 1995; Martin 1995). These 14 classifiers were each run using the yearly and quarterly feature sets, resulting in 28 classifiers in total: 14 yearly context-based classifiers and 14 quarterly context-based classifiers.

Stacked Generalization

In the third component of the framework, we utilized stacked generalization to improve procedural bias, where the classifications from the underlying individual classifiers were used as input features for a top-level classifier (Brazdil et al. 2008; Hansen and Nelson 2002; Hu and Tsoukalas 2003). All 14 classifiers described in the previous section were run as top-level classifiers, resulting in 14 different stack arrangements. Stacking can be highly effective when incorporating large quantities of predictions from underlying classifiers as input features (Lynam and Cormack 2006). Accordingly, for each stack, we utilized all 28 individual classifiers as inputs for the top-level classifier: 14 yearly context-based classifiers and 14 quarterly context-based classifiers.

The testing data for the top-level classifiers was composed of the individual (i.e., bottom-level) classifiers' classifications on the testing instances. The training data for the top-level classifiers was generated by running the bottom-level classifiers using 10-fold cross-validation on the training instances (Dzeroski and Zenko 2004; Ting and Witten 1997). In other words, the training data was split into 10 segments. In each fold, a different segment was used for testing, while the remaining 9 segments were used to train the bottom-level classifiers. The bottom-level classifiers' classifications from the test instances associated with these 10 secondary folds collectively constituted the top-level classifiers' training data. This approach was necessary to ensure that feature values in the training and testing instances of the stack classifiers were consistent and comparable (Abbasi and Chen 2009; Witten and Frank 2005).

Adaptive Learning

The ability of adaptive learning approaches to dynamically improve procedural bias are a distinct advantage of meta-learning (Brazdil et al. 2008), particularly for complex and evolving problems such as fraud detection (Fawcett and Provost 1997). We propose an adaptive semi-supervised learning (ASL) algorithm that uses the underlying generalized stacks. ASL is designed to exploit the information provided by the stack classifiers in a dynamic manner; classifications are revised and improved as new information becomes available. When semi-supervised learning is used, a critical problem arises when misclassified instances are added to the training data (Tian et al. 2007). This is a major concern in the context of financial fraud detection, where models need to be updated across years (i.e., semi-supervised active learning), since classification models can incorporate incorrect rules and assumptions, resulting in amplified error rates over time. ASL addresses this issue in two ways. First, the expansion process is governed by the stack classifiers' predictions. Only test instances that have strong prediction agreement across the top-level classifiers in the generalized stacks are added to the training data. Second, during each iteration, the training data set is reset and all testing instances are reclassified in order to provide error correction.

A high-level description of ASL's steps is as follows:

1. Train the bottom-level classifiers and run them on the entire test bed.

2. Train the top-level classifiers in the generalized stack, using the training and testing data generated by the bottom-level classifiers.

3. Reset the training data to include only the original training instances.

4. Rank the test instances based on the top-level classifiers' predictions.

5. If the stopping rule has not been satisfied, add the *d* test instances with the highest rank to the training data (with class labels congruent with the top-level classifiers' predictions) and increment *d*. Otherwise go to step 7.

6. If *d* is less than the number of instances in the test bed, repeat steps 1–5, using the expanded training data for steps 1 and 2.

7. Output the predictions from the top-level classifiers in the generalized stacks.

```
Given training examples $T = [t_1, t_2, \ldots, t_n]$, training class labels $L = [ll_1, l_2, \ldots, l_{\mathrm{n}}]$, and testing instances $R = [r_1, r_2, \ldots, r_m]$
Let $c$ denote the number of classification algorithms utilized (in this study, $c = 14$)
Initialize variable $d$ to track number of items from $R$ to add to $T$, where $p$ is a predefined constant and $d = p$
While $d \leq m$
    Derive yearly classifiers' test prediction matrix $Y = [y_1 = Yearly_1(T, L, R), \ldots, y_c = Yearly_c(T, L, R)]$ and training data cross-
        validation prediction matrix $W = [w_1 = Yearly_1(T, L), \ldots, w_c = Yearly_c(T, L)]$
    Derive quarterly classifiers' test prediction matrix $QW = [q_1 = Quarterly(T, L, R), \ldots, q_c = Quarterly_c(T, L, R)]$ and training data
        cross-validation prediction matrix $V = [v_1 = Quarterly_1(T, L), \ldots, v_c = Quarterly_c(T, L)]$
    Derive top-level stack classifiers' prediction matrix $S = [s_1 = Stack_1([W, V], L, [Y, Q]), \ldots, s_c = Stack_c([W, V], L, [Y, Q])]$
    Reset training data to original set of instances $T = [t_1, t_2, \ldots, t_n]$, training class labels $L = [l_1, l_2, \ldots, l_n]$
    Compute test instance prediction scores $P = \left[ p_1 = \sum_{i=1}^{c} s_{c1}, \ldots, p_m = \sum_{i=1}^{c} s_{cm} \right]$
    Compute test instance weights $X = [x_1, \ldots, x_m]$ where $x^i = \begin{cases} 1, if \left| p_i \right| = c \\ 0, otherwise \end{cases}$
    If $\sum_{i=1}^{m} x_i \geq d$
        Compute descending order rank of values in $X$:
        $J = [j_1 = \arg\max(a_0 = X), j_2 = \arg\max(a_1 = a_0(1{:}j_1 - 1, j_1 + 1{:}m)), \ldots, j_{\mathrm{m}} = \arg\max(a_{\mathrm{m-1}})]$
        Set the $d$ test instances with the highest rank $V = [v_1 = j_1, \ldots, v_d = j_d]$
        Determine the instances' class labels $Z = [z_1, \ldots, z_d]$ where $z_i = \begin{cases} 1, if\ p_{vi} > 0 \\ 0, otherwise \end{cases}$
        Add selected test instances to training data $T = [t_1, t_2, \ldots, t_n, V]$, $l = [l_1, l_2, \ldots, l_n, Z]$
        Increment selection quantity variable $d = d + p$
    Else
        Exit Loop
    End If
Loop
Output $S$
```

Figure 3. Adaptive Semi-Supervised Learning Algorithm

During each iteration, the training set is used to train (and run) the bottom-level classifiers on the entire test bed. The top-level classifiers are then run using the training and testing instance feature values generated by the bottom-level classifiers. The testing instances are ranked based on the top-level classifiers' predictions, where instances with greater prediction agreement across the classifiers are given a higher rank. The selected instances are added to the original training data, where the number of instances added is proportional to the iteration number (i.e., an increasing number of test instances are added during each subsequent iteration). Test instances are added with the predicted class label (as opposed to the actual label), since we must assume that the actual labels of the test instances are unknown (Chapelle et al. 2006). The instances added in one iteration are not carried over to the next one. The steps are repeated until all testing instances are added during an iteration or the stopping rule has been reached.

Figure 3 shows the detailed mathematical formulation of the ASL algorithm. In each iteration, the yearly and quarterly context-based classifiers are run with the training data T and class labels Y. These yearly and quarterly classifiers are each run in two ways. First, they are trained on T and run on the testing data R to generate the two $m \times c$ test data prediction matrices Y and Q. Next, they are run on T using 10-fold cross validation in order to generate the two $n \times c$ training data matrices (W and V) for the generalized stacks' top-level classifiers (as described earlier). The predictions from the top-level classifiers are used to construct the stack prediction matrix S. Once the stack predictions have been made, the training set is reset to its original instances in order to allow error correction in subsequent iterations in the event that an erroneous classification has been added to the training set. Next, the top-level classifiers' predictions for each instance are aggregated across classifiers (in P), and only those instances with unanimous agreement (i.e., ones deemed legitimate or fraudulent by all top-level classifiers) are given a weight of 1 in X. If the number of instances in X with a value of 1 is greater than or equal to the selection quantity variable d, we add d of these test instances to our training set T with class labels that correspond to the top-level classifiers' predictions (Z). We then increment d so that a larger number of instances will be added in the following iteration. If there are insufficient unanimous agreement instances in X, we do not continue since adding ones where the top-level classifiers disagree increases the likelihood of inserting misclassified instances into the training set. Otherwise, the process is repeated until all testing instances have been added to the training set (i.e., $d > m$).

Evaluation

Consistent with Hevner et al. (2004), we rigorously evaluated our design artifact. We conducted a series of experiments to assess the effectiveness of our proposed financial fraud detection framework; each assessed the utility of a different facet of the framework. Experiment 1 evaluated the proposed yearly and quarterly context-based feature sets in comparison with a baseline features set composed of annual statement-based financial ratios (H1). Experiment 2 assessed the effectiveness of using stacked classifiers. We tested the efficacy of combining yearly and quarterly information over using either information level alone (H2) and also compared stacked classifiers against individual classifiers (H3). Experiment 3 evaluated the performance of adaptive learning versus a static learning model (H4). Experiments 4 and 5 assessed the overall efficacy of the proposed meta-learning framework in comparison with state-of-the-art financial fraud detection methods (H5) and existing ensemble semi-supervised learning techniques (H6).

We tested the hypotheses using a test bed derived from publicly available annual and quarterly financial statements. The test bed encompassed 9,006 instances (815 fraudulent and 8,191 legitimate), where each instance was composed of the information for a given firm, for a particular year. Hence, for each instance in the test bed, the 12 financial ratios (described earlier in the section "Financial Fraud Detection Feature Sets") were derived from the annual and quarterly financial statements for that year.

The data collection approach undertaken was consistent with the approaches employed in previous studies (e.g., Cecchini et al. 2010; Dechow et al. 2011). The fraudulent instances were identified by analyzing all of the SEC Accounting and Auditing Enforcement Releases (AAERs) posted between 1995 and 2010. Based on these AAERs, fraudulent instances for the fiscal years ranging from 1985 to 2008 were identified. The information gathered from the AAERs was verified with other public sources (e.g., business newspapers) to ensure that the instances identified represented bona fide financial statement fraud cases. Consistent with prior research, firms committing fraud over a two-year period were treated as two separate instances (Cecchini et al. 2010; Dikmen and Küçükkocaoğlu 2010; Persons 1995). Thus, the 815 fraudulent instances were associated with 307 distinct firms.

The legitimate instances encompassed all firms from the same industry-year as each of the fraud instances (Beneish 1999a; Cecchini et al. 2010). After removing all non-fraud firm-year instances in which amendments/restatements had been filed as well as ones with missing statements (Cecchini et al. 2010), 8,191 legitimate instances resulted. As noted by prior studies, although the legitimate instances included did not appear in any AAERs or public sources, there is no way to guarantee that none of them have engaged in financial fraud (Bay et al. 2006; Dechow et al. 2011; Kirkos et al. 2007).

Consistent with prior work (Cecchini et al. 2010), the test bed was split into training and testing data based on chronological order (i.e., firm instance years). All instances prior to 2000 were used for training, while data from 2000 onward was used for testing. The training data was composed of 3,862 firm-year instances (406 fraudulent and 3,456 legitimate), while the testing data included 5,144 firm-year instances (409 fraudulent and 4,735 legitimate). All 14 classifiers described in the section "Yearly and Quarterly Context-Based Classifications" were employed. For all experiments, the classifiers were trained on the training data and evaluated on the 5,144-instance test set.

For financial fraud detection, the error costs associated with false negatives (failing to detect a fraud) and false positives (considering a legitimate firm fraudulent) are asymmetric. Moreover, these costs also vary for different stakeholder groups. For investors, prior research has noted that investing in a fraudulent firm results in losses attributable to decreases in stock value when the fraud is discovered, while failing to invest in a legitimate firm comes with an opportunity cost (Beneish 1999a). Analysis has revealed that the median drop in stock value attributable to financial fraud is approximately 20 percent, while the average legitimate firm's stock appreciates at a rate of 1 percent, resulting in an investor cost ratio of 1:20 (Beneish 1999a, 1999b; Cox and Weirich 2002). From the regulator perspective, failing to detect fraud can result in significant financial losses (Albrecht, Albrecht, and Dunn 2001; Dechow et al. 2011). On the other hand, false positives come with unnecessary audit costs. According to the Association of Certified Fraud Examiners (2010), the median loss attributable to undetected financial statement fraud is $4.1 million (i.e., cost of false negatives), while the median audit cost (i.e., cost of false positives) is $443,000 (Charles et al. 2010). For regulators, this results in an approximate cost ratio of 1:10. Accordingly, in this study we used cost ratios of 1:20 and 1:10 to reflect the respective situations encountered by investors and regulators.

It is important to note that, consistent with prior work, we only consider error costs (Beneish 1999a; Cecchini et al. 2010). In the case of the regulator setting, the cost breakdown is as follows:

- True Negatives: Legitimate firms classified as legitimate (no error cost)
- True Positives: Fraudulent firms classified as fraudulent (no error cost since the audit was warranted)
- False Negatives: Fraudulent firms classified as legitimate (fraud-related costs of $4.1 million)
- False Positives: Legitimate firms classified as fraudulent (unnecessary audit costs of $443,000)

Due to space constraints, in the following three subsections we only reported performance results for the investor situation, using a cost ratio of 1:20. Appendices A, B, and C contain results for the regulator situation (i.e., using a cost setting of 1:10). However, in the final two subsections, when comparing MetaFraud against other methods, results for both stakeholder groups' cost settings are reported. The evaluation metrics employed included legitimate and fraud recall for the two aforementioned cost settings (Abbasi et al. 2010; Cecchini et al. 2010). Furthermore, area under the curve (AUC) was used in order to provide an overall effectiveness measure for methods across cost settings. Receiver operating characteristic (ROC) curves were generated by varying the false negative cost between 1 and 100 in increments of 0.1, while holding the false positive cost constant at 1 (e.g., 1:1, 1:1.1, 1:1.2, etc.), resulting in 991 different cost settings. For each method, AUC was derived from these ROC curves. Moreover, all hypothesis testing results in the paper also incorporated multiple cost settings (including the investor and regulator settings).

Comparing Context-Based Classifiers against Baseline Classifiers

Most prior studies have utilized 8 to 10 financial ratios devoid of organizational or industry-level context information (e.g., Beneish 1999a; Dikmen and Küçükkocaoğlu 2010; Lin et al. 2003; Persons 1995; Spathis 2002). Accordingly, we evaluated the effectiveness of the yearly and quarterly context-based feature sets (described earlier and in Tables 3 and 4) in comparison with a baseline feature set composed of the 12 ratios described earlier. For the baseline, these 12 ratios were derived from the annual statements, as done in prior research (Kaminski et al. 2004; Kirkos et al. 2007; Summers and Sweeney 1998). The three feature sets were run using all 14 classifiers described in the previous section.

Table 5 shows the results for the baseline classifiers. Tables 6 and 7 show the results for the yearly and quarterly context-based classifiers (i.e., the 14 classifiers coupled with the 84 and 336 yearly and quarterly context-based features, respectively). Due to space limitations, we report only the overall AUC and legitimate/fraud recall (shaded columns) and legitimate/fraud precision when using the 1:20 investor cost setting (Cecchini et al. 2010). Results for the regulator cost setting can be found in Appendix A.

For all three feature sets, the best AUC results were attained using NBTree and Logit. These methods also provided the best balance between legitimate/fraud recall rates for the investor cost setting. In comparison with the baseline classifiers, the yearly and quarterly context-based classifiers had higher overall AUC values, with an average improvement of over 10 percent. For the quarterly and yearly context-based classifiers, the most pronounced gains were attained in terms of fraud recall (17 percent and 23 percent higher on average, respectively). The results for the various yearly and quarterly context-based classifiers were quite diverse: six of the quarterly classifiers had fraud recall rates over 80 percent while three others had legitimate recall values over 90 percent. Classifiers such as SVM-Linear and REPTree were able to identify over 85 percent of the fraud firms (but with false positive rates approaching 40 percent). Conversely, tree-based classifiers such as J48, Random Forest, and NBTree had false positive rates below 10 percent, but with fraud recall rates of only 25 to 50 percent. This diversity in classifier performance would prove to be highly useful when using stacked generalization (see the subsection "Evaluating Stacked Classifiers").

Context-Based Classifiers Versus Baseline Classifiers

We conducted paired t-tests to compare the performance of our yearly and quarterly context-based classifiers against the baseline. Consistent with H1, three different settings were evaluated at the yearly and quarterly level: organizational, industry, and organizational plus industry (i.e., the context classifiers from Tables 6 and 7). The t-tests were conducted

Table 5. Baseline Results Using 12 Yearly Ratios

		Legit		Fraud				Legit		Fraud	
Classifier	**AUC**	**Prec.**	**Rec.**	**Prec.**	**Rec.**	**Classifier**	**AUC**	**Prec.**	**Rec.**	**Prec.**	**Rec.**
SVM-Lin	0.522	91.3	59.9	6.8	34.0	ADTree	0.612	94.1	55.0	10.7	57.5
LogitReg	**0.703**	96.1	67.0	16.6	65.0	RandForest	0.695	95.2	67.5	14.8	59.2
J48	0.664	95.2	68.5	14.9	58.7	NBTree	0.687	95.8	84.5	30.0	54.8
BayesNet	0.673	95.5	74.0	18.2	58.2	REPTree	0.606	94.6	65.5	13.3	54.0
NaiveBayes	0.551	92.4	51.6	8.4	49.6	JRip	0.557	94.1	80.0	17.5	39.6
SVM-RBF	0.452	89.8	64.7	3.5	14.7	NNge	0.665	93.1	72.0	11.2	33.3
SVM-Poly	0.538	91.3	52.0	7.1	38.9	NeuralNet	0.591	91.9	56.0	7.8	42.8

Table 6. Yearly Context-Based Classifiers Using Organizational and Industry Context Features

		Legit		Fraud				Legit		Fraud	
Classifier	**AUC**	**Prec.**	**Rec.**	**Prec.**	**Rec.**	**Classifier**	**AUC**	**Prec.**	**Rec.**	**Prec.**	**Rec.**
SVM-Lin	0.694	99.3	36.9	11.7	96.8	ADTree	0.773	97.7	58.5	14.9	84.1
LogitReg	0.791	97.0	70.1	17.8	74.8	RandForest	0.785	96.4	71.3	17.2	69.2
J48	0.669	94.1	82.0	16.3	40.6	NBTree	**0.814**	96.6	73.6	18.6	69.7
BayesNet	0.752	96.5	73.4	18.3	68.7	REPTree	0.624	96.0	61.1	13.5	70.2
NaiveBayes	0.716	98.2	55.2	14.6	88.5	JRip	0.626	95.3	69.7	14.6	60.2
SVM-RBF	0.645	94.2	63.3	11.5	55.3	NNge	0.703	94.2	70.3	12.7	50.1
SVM-Poly	0.729	98.5	52.5	14.2	90.7	NeuralNet	0.619	95.9	59.2	13.1	70.9

Table 7. Quarterly Context-Based Classifiers Using Organizational and Industry Context Features

		Legit		Fraud				Legit		Fraud	
Classifier	**AUC**	**Prec.**	**Rec.**	**Prec.**	**Rec.**	**Classifier**	**AUC**	**Prec.**	**Rec.**	**Prec.**	**Rec.**
SVM-Lin	0.733	98.7	60.3	16.5	90.7	ADTree	0.741	98.0	53.9	14.1	87.3
LogitReg	**0.780**	96.5	69.4	16.6	70.7	RandForest	0.689	93.6	93.4	25.0	25.7
J48	0.739	95.5	90.8	31.9	49.9	NBTree	0.724	93.7	92.3	24.1	28.4
BayesNet	0.645	95.6	67.4	14.5	64.1	REPTree	0.761	97.9	52.4	13.6	86.8
NaiveBayes	0.724	96.7	50.7	12.3	80.2	JRip	0.670	95.6	63.3	13.4	66.0
SVM-RBF	0.745	98.2	60.3	16.0	87.5	NNge	0.703	93.2	80.6	12.3	31.5
SVM-Poly	0.742	97.1	55.0	13.4	80.7	NeuralNet	0.652	94.4	65.6	12.1	55.0

on legitimate and fraud recall. We ran paired t-tests using cost settings of 1:5 to 1:50 (in increments of 5) across the 14 classifiers. For each test, this resulted in 140 controlled pairs: 10 cost settings multiplied by 14 classifiers. This particular range of costs was chosen since it incorporated the two cost settings used in this study as well as those employed in prior studies (e.g., Beneish 1999a; Cecchini et al. 2010), and therefore represented a reasonable range of costs for various stakeholder groups. Costs and classifiers were controlled in the t-test in order to isolate the impact of the input features (as described in H1).

Table 8 shows the t-test results. The three yearly settings each significantly outperformed the baseline classifiers on fraud recall/precision and legit precision. However, the performance gains for legitimate recall were not significant for alpha set to 0.05, with p-values of 0.211, 0.056, and 0.065. This was due to the imbalanced performance of the baseline ratios (high legit recall and very low fraud recall). Nevertheless, the yearly organizational, industry, and context feature sets all improved legitimate recall (with the latter two significant at alpha = 0.1). The three quarterly settings significantly outperformed the baseline classifiers in terms of legitimate

Table 8. P-Values for Pair-Wise t-Tests for Individual Classifiers Versus Baseline (n = 140)

	Yearly Ratio Hypotheses			Quarterly Ratio Hypotheses		
Metric	H1a: Org.	H1b: Industry	H1c: Context	H1d: Org.	H1e: Industry	H1f: Context
Legit Precision	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001
Legit Recall	0.211	0.056	0.065	0.003	0.001	< 0.001
Fraud Precision	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001	< 0.001
Fraud Recall	< 0.001	< 0.001	< 0.001	0.014	0.016	0.026

and fraud recall. Overall, 21 out of 24 conditions were significant at alpha = 0.05, while 23 out of 24 were significant at alpha = 0.1. The results support H1a through H1f: feature sets composed of financial ratios as well as organizational and industry-level context measures are more adept at detecting financial fraud than ones solely employing financial ratios.

To further assess the impact of the context features, we computed the information gain (IG) for each feature. IG is a univariate measure of the amount of entropy reduction provided by a particular attribute across classes. IG scores are greater than or equal to 0, with higher scores indicating greater discriminatory potential. Related fraud detection work has used IG to measure the ability of context features to discriminate between fraud and non-fraud instances (Abbasi et al. 2010). We computed the IG scores for all features in the yearly and quarterly context-based feature sets. Table 9 shows the top 15 features (based on their IG score) in both feature sets. For each feature, the three columns denote the feature rank, description, and IG value, respectively. For the description columns, the letters R, T, C, Q, and P represent ratio, top-5 industry model, closest-5 industry model, quarter, and previous year, respectively, while the numbers indicate the ratio or quarter number. Hence, for the yearly context-based features, the description R2 denotes the second ratio in the subsection "Financial Ratios." R7 / P7 represents ratio number 7 divided by ratio number 7 from the previous year. R2 – T2 signifies the value of ratio number 2 (from the subsection "Financial Ratios") minus the value of ratio 2 for the firms' corresponding top-5 industry model. The top ranked quarterly ratio-based feature R8Q4 / C8Q4 signifies the value of ratio 8 (from "Financial Ratios") in quarter 4 divided by the value of ratio 8 in quarter 4 for the firms' corresponding closest-5 industry model. Based on Table 9, it is apparent that most of the best features were industry-level and organizational context-based measures. These context measures, coupled with ratios 1, 7, and 8 (asset quality index, inventory growth, and leverage) seemed to provide the best discriminatory potential. The table provides insights into how the context-based measures supplemented the financial ratios, resulting in enhanced financial fraud detection capabilities

Evaluating Stacked Classifiers

We evaluated the effectiveness of using stacking, where the classifications from the underlying individual classifiers were used as input features for the top-level classifiers. Three types of stacks were utilized: yearly, quarterly, and combined. The top-level yearly stack classifiers each used the 14 yearly context-based classifiers as their input features. The top-level quarterly stack classifiers each used the 14 quarterly context-based classifiers, while the top-level combined stack classifiers used all 28 yearly and quarterly context-based classifiers as input features. For each of three categories of stacks, we used all 14 classification methods as the top-level classifiers. The stack classifiers were trained and tested using the approach described in the subsection "Stacked Generalization." The training data for the top-level classifiers was generated by running the bottom-level classifiers using 10-fold cross-validation on the training instances. The resulting classifications generated by the bottom-level classifiers for all the training instances collectively constituted the top-level classifiers' training data.

Tables 10 and 11 show the results for the yearly and quarterly stack classifiers. Most stacks had AUC values above 0.8, with the highest value attained using quarterly stack with an SVM-RBF top-level classifier. For the investors' cost situation, fraud recall rates tended to be considerably higher than legitimate recall rates. This outcome, which is a result of the cost setting used (i.e., 1:20), is likely desirable since an investor's costs associated with failing to detect fraud are considerably higher than the costs associated with false positives (Beneish 1999a). The yearly and quarterly stack classifiers' results were noticeably better than those achieved using their individual context-based counterparts (see Table 6), with an average gain of over 11 percent in terms of AUC. All 14 yearly stacks had higher fraud f-measures, while 13 also had higher legitimate f-measures, suggesting that in addition to improved overall accuracy, the yearly stacks were far more balanced in their performance across legitimate and fraudulent instances. For the investor cost setting, the most balanced results were attained using a Random Forest classifier. The

Table 9. IG Measures for Top 15 Attributes in Yearly and Quarterly Context-Based Feature Sets

Yearly Context-Based Features			Quarterly Context-Based Features		
Rank	**Description**	**IG Value**	**Rank**	**Description**	**IG Value**
1	R2	0.0276	1	R8Q4 / C8Q4	0.0619
2	R7 / P7	0.0249	2	R7Q2	0.0505
3	R3	0.0249	3	R7Q3 - C7Q3	0.0503
4	R9	0.0234	4	R8Q3 - T8Q3	0.0499
5	R2 - T2	0.0231	5	R7Q1 / T7Q1	0.0489
6	R8 - C8	0.0193	6	R8Q4 – R8Q3	0.0486
7	R2 - C2	0.0188	7	R1Q3 – R1Q2	0.0482
8	R1 / P1	0.0182	8	R8Q2 / C8Q2	0.0480
9	R7 - T7	0.0173	9	R8Q3 / T8Q3	0.0474
10	R8 - P8	0.0168	10	R8Q1 / R8PQ4	0.0473
11	R7	0.0168	11	R8Q1 / C8Q1	0.0470
12	R7 / T7	0.0163	12	R7Q1 – C7Q1	0.0458
13	R1 - C1	0.0162	13	R8Q1 - C8Q1	0.0452
14	R8 / T8	0.0149	14	R12Q3 – C12Q3	0.0441
15	R8 - T8	0.0147	15	R8Q4 / T8Q4	0.0427

Table 10. Yearly Stack Classifiers

		Legit		Fraud				Legit		Fraud	
Classifier	**AUC**	**Prec.**	**Rec.**	**Prec.**	**Rec.**	**Classifier**	**AUC**	**Prec.**	**Rec.**	**Prec.**	**Rec.**
SVM-Lin	0.843	98.2	68.7	19.0	85.1	ADTree	0.823	97.9	73.0	20.6	81.4
LogitReg	0.827	97.9	72.0	20.2	81.9	RandForest	0.836	96.9	81.1	24.1	69.4
J48	0.828	97.1	74.2	19.8	73.8	NBTree	**0.863**	98.5	72.9	21.8	87.3
BayesNet	0.850	97.2	76.3	21.4	74.6	REPTree	0.812	97.2	71.8	18.8	75.8
NaiveBayes	0.814	97.1	81.2	24.9	72.1	JRip	0.817	97.0	79.4	23.1	71.6
SVM-RBF	0.853	97.9	72.0	20.2	82.4	NNge	0.827	96.7	77.0	20.8	69.7
SVM-Poly	0.797	97.5	71.7	19.3	78.5	NeuralNet	0.823	97.5	73.9	20.6	78.2

Table 11. Quarterly Stack Classifiers

		Legit		Fraud				Legit		Fraud	
Classifier	**AUC**	**Prec.**	**Rec.**	**Prec.**	**Rec.**	**Classifier**	**AUC**	**Prec.**	**Rec.**	**Prec.**	**Rec.**
SVM-Lin	0.864	98.4	66.6	18.4	87.3	ADTree	0.838	99.8	66.6	20.2	98.0
LogitReg	0.837	98.3	70.5	20.1	85.6	RandForest	0.856	97.8	75.2	21.9	80.4
J48	0.821	98.5	68.5	19.5	88.3	NBTree	0.873	99.0	65.9	19.0	92.4
BayesNet	0.874	98.8	67.6	19.4	90.2	REPTree	0.866	99.3	65.9	19.4	94.9
NaiveBayes	0.805	97.7	67.6	17.9	81.9	JRip	0.837	99.2	66.9	19.7	93.9
SVM-RBF	**0.881**	99.5	69.9	21.6	96.1	NNge	0.805	98.0	72.1	20.4	82.6
SVM-Poly	0.862	98.9	66.6	19.1	91.2	NeuralNet	0.794	98.0	67.2	18.2	84.4

Table 12. Combined Stack Classifiers

Classifier	AUC	Legit		Fraud		Classifier	AUC	Legit		Fraud	
		Prec.	Rec.	Prec.	Rec.			Prec.	Rec.	Prec.	Rec.
SVM-Lin	0.904	98.0	80.0	26.0	81.2	ADTree	0.896	99.2	76.0	25.0	92.4
LogitReg	0.875	97.8	84.9	30.9	78.2	RandForest	0.857	96.7	95.2	53.0	62.8
J48	0.839	97.2	77.6	22.3	74.3	NBTree	0.887	98.8	77.0	25.0	89.0
BayesNet	0.893	98.1	77.9	24.3	82.2	REPTree	0.888	98.6	75.3	23.5	88.0
NaiveBayes	0.851	97.4	90.6	39.8	71.6	JRip	0.870	98.8	80.2	27.9	88.8
SVM-RBF	0.894	98.9	74.5	23.3	90.0	NNge	0.865	96.5	88.4	32.1	63.3
SVM-Poly	0.865	97.5	88.9	36.5	73.8	NeuralNet	0.865	98.7	80.3	27.8	88.0

Table 13. P-Values for Pair-Wise t-Tests for Combined Versus Yearly/Quarterly Stack Classifiers (n = 140)

Metrics	Legit		Fraud	
	Precision	Recall	Precision	Recall
H2a: Combined versus Yearly	< 0.001	< 0.001	< 0.001	< 0.001
H2b: Combined versus Quarterly	< 0.001	0.001	< 0.001	< 0.001[†]

[†] Opposite to hypothesis

quarterly stack classifiers had considerably better fraud recall rates than the individual quarterly context-based classifiers (see Table 7), outperforming them by nearly 25 percent on average. However, this improved fraud recall was coupled with reduced legitimate recall for four of the classifiers.

When compared to the yearly stacks, the quarterly stacks yielded higher fraud recall rates (12 percent higher on average); every quarter stack classifier identified at least 80 percent of the fraud firms, albeit with false positive rates generally above 30 percent. However, both the yearly and quarterly stacks had comparable average AUC values. This was attributable to the yearly stacks' 6 percent higher average performance on legitimate firms, which resulted in considerably lower false positive rates.

Table 12 shows the results for the combined stack classifiers, which used the yearly and quarterly context-based classifiers' classifications as input. The overall AUC values ranged from 0.839 to 0.904, with the best results attained using a SVM-Linear classifier. All 14 combined stacks outperformed their yearly and quarterly stack counterparts, with an average AUC gain of 3 to 5 percent. For the investor cost setting, while some of the combined stacks had higher legitimate/fraud recall rates, they were generally more balanced than the yearly and quarterly stacks in terms of their classifications (e.g., SVM-Linear, Logistic Regression, J48, Bayesian Network, JRip, and Neural Network). This suggests that the enhanced performance of the combined stacks was attributable to their ability to leverage the complementary information provided by the yearly and quarterly context-based classifiers.

H2: Combined Stack Classifiers Versus Yearly and Quarterly Stack Classifiers

We conducted paired t-tests to compare the performance of the combined stacks against the yearly (H2a) and quarterly (H2b) stack classifiers using the same setup as the previous hypothesis test (H1). Table 13 shows the t-test results. The combined stack classifiers significantly outperformed the yearly stack classifiers on legitimate and fraud precision/recall (with all four p-values < 0.01). The combined stacks also outperformed the quarterly stacks on legitimate precision/recall and fraud precision. However, the quarter stacks had significantly better performance gain on fraud recall (denoted by a plus sign). Overall, seven out of eight tests were significant; the t-test results support H2a–b and suggest that combining yearly and quarterly information can facilitate enhanced financial fraud detection capabilities.

To further assess the effectiveness of combining yearly and quarterly information, we performed an instance-level analysis of the classifications made by all 14 combined stack classifiers. For each of the instances in our test bed, we aggregated

Sum of Stack Scores|	Legit			Fraud			Cumulative% of Total Inst.
	% Prec.	% Rec.	% Inst.	% Prec.	% Rec.	% Inst.	
14	99.4	96.5	33.4	61.8	90.8	24.0	32.7
12	99.4	94.6	14.4	66.4	94.8	18.8	47.5
10	98.2	93.5	12.6	67.8	89.1	22.5	60.9
8	98.5	89.6	9.5	48.4	88.0	12.2	70.6
6	98.9	84.2	8.7	27.0	85.7	6.9	79.2
4	97.7	80.3	9.0	20.0	72.4	7.1	88.0
2	94.6	65.9	6.1	13.2	57.7	6.4	94.2

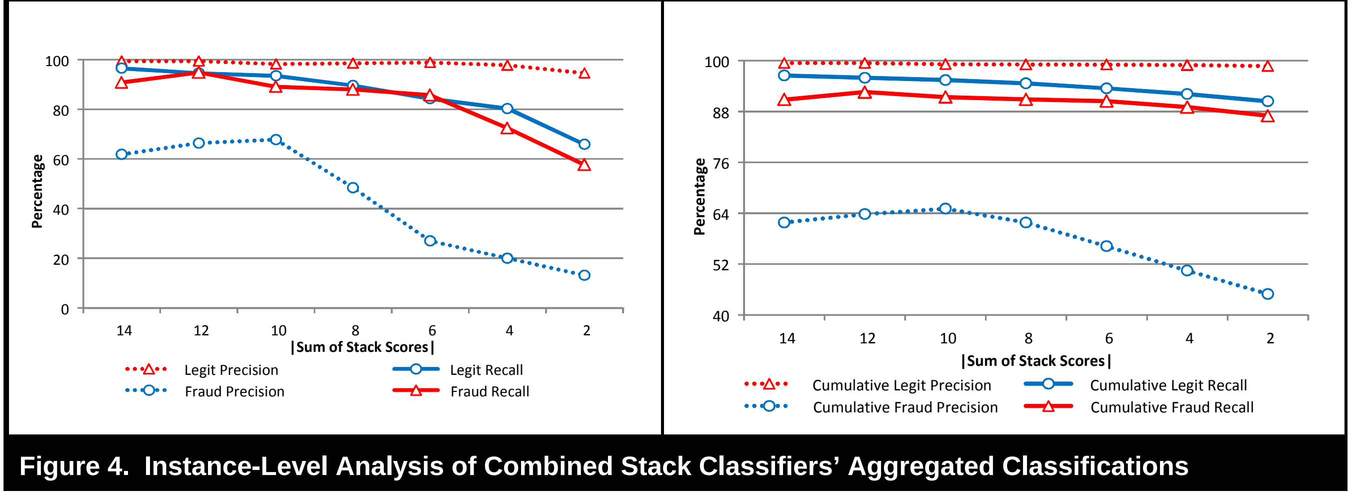


Figure 4. Instance-Level Analysis of Combined Stack Classifiers' Aggregated Classifications

that particular instance's 14 classification scores. Since the classifiers assigned a "-1" to instances classified as fraudulent and "1" to instances considered legitimate, the aggregated scores varied from -14 to 14 (in intervals of 2). Hence, a score of 12 meant that 13 classifiers considered the instance legitimate and 1 deemed it fraudulent. The absolute value $|x|$ of an aggregated score x represents the degree of agreement between stack classifiers. For each $|x|$, class-level precision and recall measures were computed as follows: All instances with positive aggregated scores (i.e., $x > 0$) were considered legitimate (i.e., scores of 2, 4, 6, 8, 10, 12, and 14) while those with negative scores (i.e., $x < 0$) were considered fraudulent. These score-based predictions were compared against the actual class labels to generate a confusion matrix for each $|x|$. From each confusion matrix, legit/fraud precision and recall were computed, resulting in performance measures for every level of classifier agreement (i.e., minimal agreement of $|x| = 2$ to maximal agreement $|x| = 14$). It is important to note that the 5.8 percent of instances in the test bed that had an aggregated score of 0 were excluded from the analysis (i.e., test instances where the classifiers were evenly split between predictions of legitimate and fraudulent).

Figure 4 shows the analysis results. The first column shows $|x|$; the absolute value of the aggregated score x for an instance. Columns two, three, five, and six show legitimate/fraud precision and recall percentages attained on instances with that particular score. Columns four, seven, and eight depict the percentage of the legit, fraud, and total instances in the test bed covered by that score, respectively. The chart at the bottom left shows plots of the precision rates (i.e., columns 2 and 5) and recall rates (columns 3 and 6) for instances with that score. The chart at the bottom right shows cumulative precision and recall rates if using that score as a threshold. The results can be interpreted as follows: 32.7 percent of all instances in the test set had a score of -14 or 14. These instances accounted for 33.4 percent of all legitimate test instances and 24 percent of all fraud test instances. Of these instances, 96.5 percent of the legitimate instances were correctly classified (i.e., $x = 14$) while the remaining 3.5 percent were misclassified as fraudulent (i.e., $x = -14$).

The results reveal that these aggregated scores provide a nice mechanism for assessing the confidence level of a particular classification. When all 14 combined stacks agreed (i.e., the

NAICS Code	Description	# Firms	% Recall
31–33	Manufacturing	129	94.57
51	Information	72	77.78
52	Finance and Insurance	57	73.68
54	Professional Services	33	87.88
42	Wholesale Trade	23	95.65
23	Construction	17	88.24
21	Mining	15	93.33
44–45	Retail Services	13	100.00
56	Waste Management	11	81.82

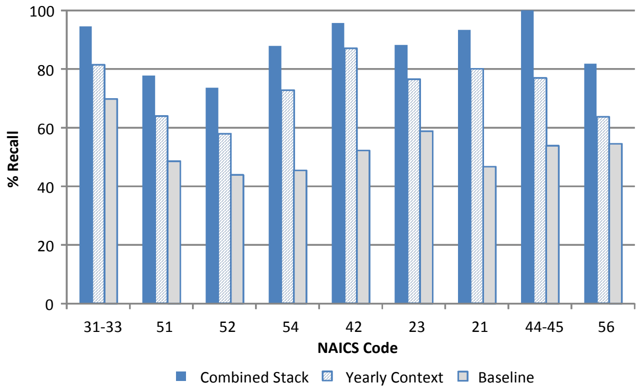


Figure 5. Analysis of Combined Stack Classifiers' Fraud Recall Rates for Various Business Sectors

absolute value of the sum of their classification scores was 14), the legit and fraud recall rates were 96.5 and 90.8 percent, respectively. Moreover, this accounted for 32 percent of the test instances. Looking at the chart on the left, as expected, lower scores generally resulted in diminished recall rates (one exception was fraud recall, which increased when using a score of 12). Based on the chart on the right, the performance degradation was quite gradual for thresholds greater than 4. For example, using a threshold of 6 or better resulted in legitimate and fraud recall rates of over 90 percent on instances encompassing 79.2 percent of the test bed. Using the aggregated scores as a confidence-level measure can be useful for prioritizing regulator or investor resources. Moreover, the stack scores can also be exploited in a semi-supervised learning manner to further improve performance, as discussed in the next subsection.

We also analyzed the combined stack classifiers' fraud detection rates for different sectors by categorizing the 409 fraud firm-years in the test set based on their top-level NAICS classifications (also referred to as business sectors). While the 20 top-level classifications (e.g., manufacturing, wholesale, retail, construction, mining, etc.) are more general than the 1,017 bottom-level industry classifications in the NAICS hierarchy, which were used to build the industry-level models, aggregating results at the top-level provided interesting insights. Figure 5 shows the results for all top-level NAICS codes with at least 10 fraud firm-years in the test set. The table on the left lists the results for the combined stack classifiers, while the chart on the right shows the combined stacks' results relative to the yearly context and baseline classifiers. The combined stacks performed best on merchandise firms at various stages of the supply chain (e.g., manufacturing, wholesale, retail), with fraud recall of between 94 and 100 percent on such firms. The yearly context and baseline classifiers also performed best on these sectors, although with lower detection rates. The enhanced performance on manufacturing firms is consistent with previous work that has also attained good fraud detection rates on such data (Kirkos et al. 2007; Spathis 2002). The combined stacks had relatively lower fraud recall rates on firms in the information and finance/insurance sectors (73 percent and 77 percent), although still higher than the yearly context (approximately 60 percent) and baseline classifiers (less than 50 percent). Analysis of the test bed revealed that fraud firms in these two sectors were 25 percent more prevalent in the test set, as compared to the training data. Since fraud is often linked to financial distress (Chen and Du 2009; Zhao et al. 2009), the increased number of fraud firms from the information sector appearing in the year 2000 onward could be related to the dot-com bubble burst. Similarly, fraud in the financial sector continues to grow and play a pivotal role in the economy (Stempel 2009). Such temporal changes in fraud patterns attributable to overall economic and industry-level conditions further underscore the importance of adaptive learning approaches (discussed in the next subsection).

H3: Yearly and Quarterly Stack Classifiers Versus Context-Based Classifiers

We conducted paired t-tests to compare the performance of the yearly stacks against the yearly context-based classifiers (H3a) and the quarterly stacks against the quarterly context-

Table 14. P-Values for Pair-Wise t-Tests for Stack Versus Individual Classifiers (n = 140)

	Legit		Fraud	
Metrics	**Precision**	**Recall**	**Precision**	**Recall**
H3a: Yearly Stack-Individual	< 0.001	< 0.001	< 0.001	0.011
H3b: Quarterly Stack-Individual	< 0.001	0.482	0.008	< 0.001

based classifiers (H3b) (see Table 14). The yearly stacks significantly outperformed the yearly context-based classifiers on legitimate and fraud precision/recall, with all four p-values less than 0.01. With respect to H3b, the quarterly stacks significantly outperformed the quarterly context-based classifiers on legitimate precision and fraud precision/recall. While the quarterly stacks also had higher legitimate recall rates, the gains were not significant. Overall, three of the four conditions were significant. The results support our hypothesis that stack classifiers can improve financial fraud detection capabilities over simply using individual classifiers, irrespective of whether yearly or quarterly features are being employed.

Figure 6 depicts the mean and range of the yearly stack (left chart) and yearly context (right chart) ROC curves. The solid black lines show the mean ROC curves (taken as the average across the 14 classifiers' curves), which depict the tradeoffs between true and false positives. Here, false positives indicate legitimate firms misclassified as being fraudulent. Curves situated closer to the top left corner signify better results, since they denote high ratios of true to false positives. The shaded areas indicate the ROC curve range (minimum and maximum values) across the 14 yearly stack and context-based classifiers. The two values on the bottom right corner of the charts indicate the mean AUC (i.e., the AUC for the solid black lines) and the mean range (i.e., the area of the shaded regions). Looking at the charts, we can see that the yearly context-based classifiers had greater variability across classifiers; the larger shaded region is indicative of classifiers that varied considerably in terms of their true/false positive rates (with some providing better legitimate recall, and others specializing in fraud recall). In contrast, the stacks provided better performance and also exhibited less variation; their curves were situated closer to one another (as evidenced by the smaller shaded region), while the results of the context-based classifiers are relatively sporadic. The AUC values varied 6 percent across the yearly stacks and 19 percent across the yearly context-based classifiers. Although not depicted here, a similar trend was observed with the quarterly stack and context classifiers. These results are consistent with prior research, which has suggested that stacked generalization can improve performance, generalization ability, and reliability when applied to a diverse set of complementary underlying classifiers (Lynam and Cormack 2006; Sigletos et al. 2005; Wolpert 1992).

Evaluating Adaptive Semi-Supervised Learning

We ran the ASL algorithm on top of the base and stack classifiers, as described in the "Adaptive Learing" subsection. During each iteration, we incremented d by 10 (i.e., $p = 10$). The test instances were evaluated in chronological order. In other words, ASL was first run on the data from 2000. Once all of the instances from 2000 had been processed (i.e., assigned final predictions), ASL moved on to the firms from 2001. Once the algorithm had finished running, the prediction values in the S matrix were used to compute the performance of the 14 classifiers. Table 15 shows the results. All 14 classifiers had AUC values above 0.866, with many classifiers attaining values over 0.900. The most balanced results were attained using an NBTree top-level classifier, which yielded the highest AUC and attained legitimate and fraud recall rates above 88 percent for the investor cost setting. The results of all 14 ASL classifiers were better than those achieved using their combined stack counterparts (i.e., static supervised learning). The biggest gains were achieved when the Bayesian Network classifier was used.

H4: ASL Versus Combined Stack Classifiers

We conducted paired t-tests (n = 140) to compare the performance of the adaptive semi-supervised learning classifiers against the combined stack classifiers (H4). The adaptive semi-supervised learning classifiers significantly outperformed the combined stack classifiers on legitimate precision/recall (both p-values < 0.001) and fraud precision/recall (p-values < 0.001 and 0.049, respectively). The t-test results support H4 and suggest that adaptive learning can further enhance financial fraud detection performance over static supervised learning models.

The two solid lines in Figure 7 show the legitimate and fraud recall performance of ASL in comparison with the combined

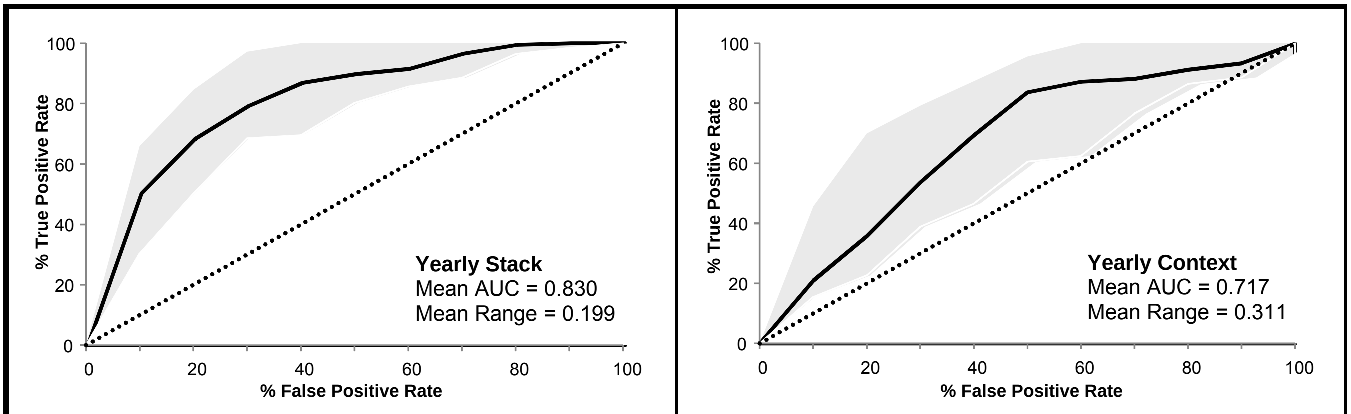


Figure 6. Mean and Range of ROC Curves for Yearly Stack and Context Classifiers

Table 15. Adaptive Semi-Supervised Learning

		Legit		Fraud				Legit		Fraud	
Classifier	AUC	Prec.	Rec.	Prec.	Rec.	Classifier	AUC	Prec.	Rec.	Prec.	Rec.
SVM-Lin	0.912	98.4	86.6	35.1	83.9	ADTree	0.904	99.2	83.6	32.7	92.2
LogitReg	0.911	98.3	88.7	38.5	81.9	RandForest	0.881	97.2	95.8	58.5	68.2
J48	0.866	97.6	89.9	38.9	74.3	NBTree	**0.922**	98.8	88.3	39.4	88.0
BayesNet	0.918	98.3	91.1	44.4	82.2	REPTree	0.893	98.5	85.0	32.8	84.6
NaiveBayes	0.872	97.4	91.9	43.4	71.6	JRip	0.905	98.6	85.9	34.5	86.1
SVM-RBF	0.906	99.0	86.7	36.8	90.0	NNge	0.884	97.2	94.6	52.1	68.7
SVM-Poly	0.895	97.6	89.0	37.0	75.1	NeuralNet	0.876	98.9	84.5	33.1	88.8

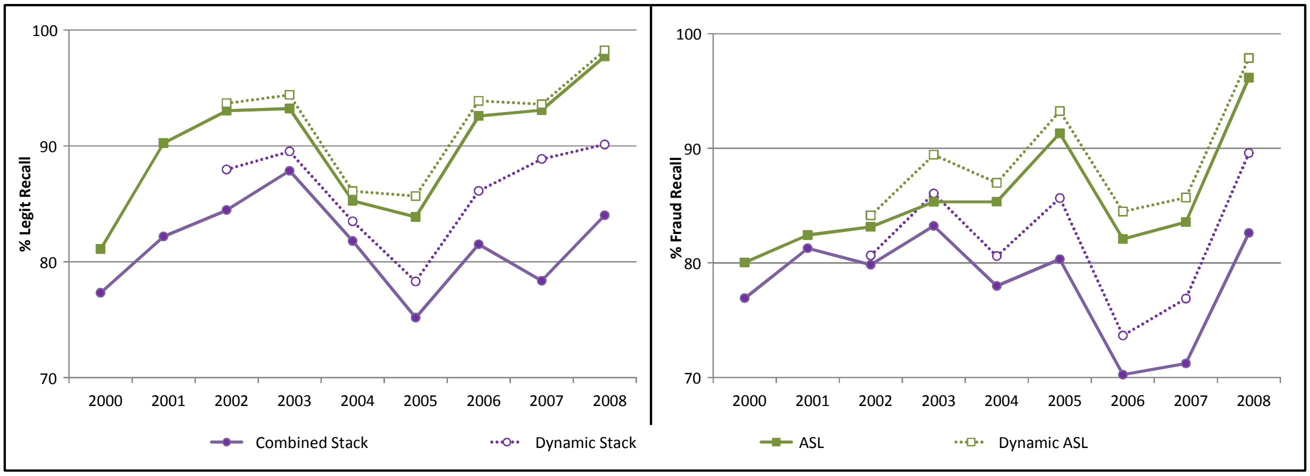


Figure 7. Performance of ASL and Combined Stack across Test Years

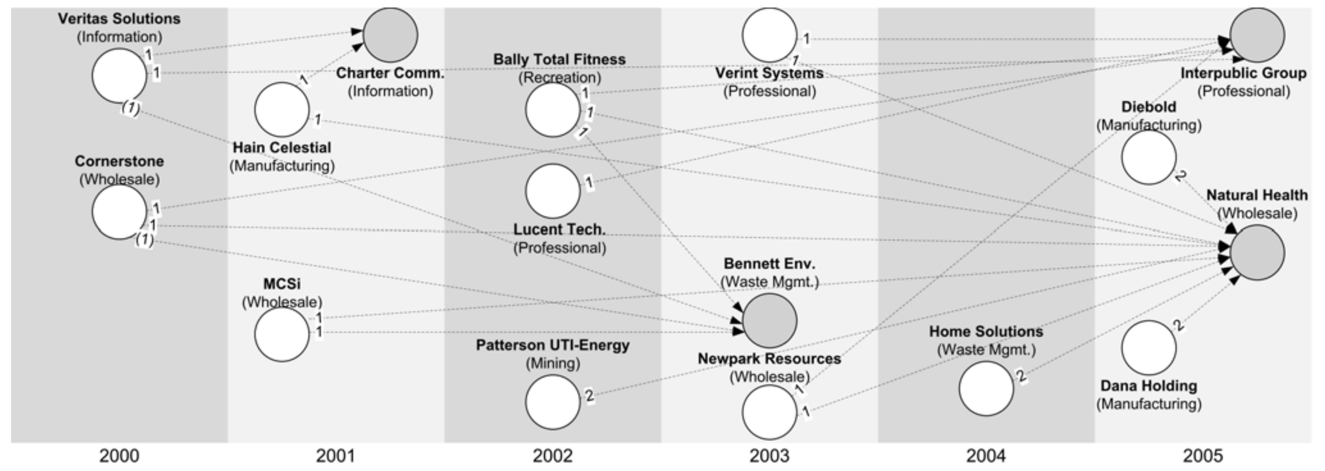


Figure 8. Illustration of Impact of Adaptive Learning on Fraud Firms Detected by ASL

stack for each year of the test bed (2000–2008). ASL had higher legit/fraud recall rates across years. Margins seemed to improve in later years. The improved performance on both legitimate and fraud firms suggests that adaptive learning is important not only to better detect fraud firms, but also in order to react to changes in non-fraud firms. The biggest improvements in fraud recall were in the information and financial and insurance sectors (over 5 percent), the two sectors where the combined stacks underperformed. Given the adversarial nature of fraud detection (Virdhagriswaran and Dakin 2006), one may expect fraud detection rates for static models to deteriorate over time, as evidenced by the decreasing performance for the combined stacks from 2004 onward. In contrast, ASL's fraud performance holds steady across these years.

However, this analysis assumes that no new training data is made available during the test time period 2000–2008 (i.e., only data through 1999 was used for training). In order to evaluate the impact of having additional training data on the performance of ASL and the combined stacks, we used an expanding window approach. For a given test year *a*, all instances up to and including year *a* – 2 were used for training (e.g., test year 2002 was evaluated using training data through 2000). A two-year window was used since prior research has noted that the median time needed to detect financial fraud is 26.5 months (Beneish 1999b). The dotted lines in Figure 7 show the results for these "dynamic" ASL and stack classifiers. Based on the results, as expected, the dynamic stacks outperformed the static combined stacks. The gains in legitimate and fraud recall were most pronounced for 2005 onward. Dynamic ASL also improved performance over ASL, particularly for fraud recall. Moreover, ASL outperformed the dynamic stack in legitimate recall for all 7 years and on fraud recall for 6 of 7 years, while dynamic ASL dominated the dynamic stack with respect to legitimate/fraud recall. Appendix F provides additional analysis which shows that ASL outperformed dynamic stacks for any window length between 1 and 5 years. The results suggest that ASL is able to effectively leverage existing knowledge, including knowledge gained during the detection process, toward enhanced subsequent detection of firms.

To illustrate this point, we analyzed fraud firm-years correctly detected by ASL that were not identified by the combined stacks (we refer to these instances as *y* firm-years). Sensitivity analysis was performed to see how fraud firms previously added to the classification models (*x* firm-years) subsequently impacted ASL's prediction scores for each of these *y* firm-years. This was done by failing to add each *x* firm-year to the model, one at a time, in order to assess their individual impact on the ASL prediction scores for the *y* firm-years. One example generated as a result of the analysis is depicted in Figure 8. The four gray nodes represent *y* firm-years: fraud instances correctly identified by ASL (that were misclassified by the combined stacks). White nodes represent *x* firm-years: fraud instances from the test set that were added to the classification models by ASL. Node labels indicate firms' names and top-level NAICS categories. The x-axis displays the years associated with these firm-years (2000–2005). A directed link from a white node to a gray one can be interpreted as that *x* firm-year influencing the prediction score of the *y* firm-year. The nature and extent of the influence is indicated by the number along the link. For example, a "2" means that the absence of that particular *x* firm-year from the classification model worsened the *y* firm-year's ASL score by two. For example, the addition of MCSi in 2002 improved the prediction score for Natural Health in 2005 by one.

Figure 8 provides several important insights regarding the adaptive learning component of the MetaFraud framework. It reveals that new fraud firm-year detections were not necessarily attained simply by adding one or two additional fraud cases. Rather, they were sometimes the result of a complex series of modifications to the training models. In some cases, these correct predictions were the culmination of modifications spanning several years of data. For example, the detection of Interpublic Group and Natural Health in 2005 leveraged 6–10 *x* firm-years between 2000 and 2005. However, firms from the same year also played an important role, as evidenced by the positive impact Diebold and Dana Holding had on Natural Health in 2005. Interestingly, business sector affiliations also seemed to play a role in the adaptive learning process: several of the links in the figure are between firms in the same sector. For example, three of the firms that influenced Natural Health were also from the wholesale trade sector, while one or two of the firms that impacted Charter Communications and Interpublic Group were from the information and professional services sectors, respectively. It is also important to note that not all *x* firm-years' additions to the models had a positive impact. For example, both Veritas Solutions and Cornerstone (from 2000) worsened Bennett Environmental's prediction score by one. Thus, Figure 8 sheds light on how ASL was able to improve the detection of fraudulent firms. It is important to note that in our analysis, consistent with prior work, we represent each fraud firm based on the year in which the fraud occurred (Cecchini et al. 2011). An interesting future direction would be to also consider when the fraud was actually discovered, and to use this information to retrospectively identify previously undetected frauds committed in earlier years.

Evaluating MetaFraud in Comparison with Existing Fraud Detection Methods

We evaluated MetaFraud in comparison with three prior approaches that attained state-of-the-art results: Kirkos et al. (2007), Gaganis (2009), and Cecchini et al. (2010). Each of these three approaches was run on our test bed, in comparison with the overall results from the proposed meta-learning framework. Kirkos et al. and Gaganis each attained good results on Greek firms, with overall accuracies ranging from 73 to 90 percent and 76 to 87 percent, respectively. Cecchini et al. attained excellent results on U.S. firms, with fraud detection rates as high as 80 percent. In comparison, prior studies using public data all had fraud detection rates of less than 70 percent. The details regarding the three comparison approaches are as follows.

Kirkos et al. used a set of 10 ratios/measures in combination with three classifiers: ID3 decision tree, neural network, and Bayesian network. We replicated their approach by including the same 10 ratios/measures: debt to equity, sales to total assets, sales minus gross margin, earnings before income tax, working capital, Altman's Z score, total debt to total assets, net profit to total assets, working capital to total assets, and gross profit to total assets. We also tuned the three classification algorithms as they did in their study (e.g., the number of hidden layers and nodes on the neural network).

Gaganis used 7 financial ratios in combination with 10 classification algorithms. We replicated his approach by including the same seven ratios: receivables to sales, current assets to current liabilities, current assets to total assets, cash to total assets, profit before tax to total assets, inventories to total assets, and annual change in sales. We ran many of his classification methods that had performed well, including neural network, linear/polynomial/RBF SVMs, logistic regression, k-Nearest neighbor, and different types of discriminant analysis. The parameters of all techniques were tuned as was done in the original study. We included the results for the three methods with the best performance: linear SVM, Neural Net, and Logit.

Cecchini et al. used an initial set of 40 variables. After preprocessing, the remaining 23 variables were used as input in their financial kernel. Following their guidelines, we began with the same 40 variables and removed 19 after preprocessing, resulting in 21 input variables for the financial kernel.

All comparison techniques were run using the same training and testing data used in our prior experiments. ROC curves were generated, and AUC for the ROC curves was computed (as with the previous experiments). For MetaFraud, we generated a final prediction for each test case by aggregating the predictions of the 14 ASL classifiers to derive a single stack score for each instance (as described earlier). We used the results of the ASL classifiers since these results are based on the amalgamation of all four phases of MetaFraud and therefore signify the final output of the meta-learning framework.

Table 16 and Figure 9 show the experiment results. Table 16 depicts the results for MetaFraud (MF) and the comparison methods using both the regulator (1:10) and investor (1:20) cost settings. In addition to legitimate and fraud recall, we

Table 16. Results for MetaFraud and Comparison Methods

Setting	AUC	Regulators (Cost 1:10)					Investors (Cost 1:20)				
		Legit		Fraud		Error Cost	Legit		Fraud		Error Cost
		Prec.	Rec.	Prec.	Rec.		Prec.	Rec.	Prec.	Rec.	
MetaFraud (MF)	0.931	98.3	90.2	41.8	81.5	$100.5	98.4	89.6	40.8	82.9	$441.1
Cecchini	0.818	97.8	79.8	25.4	79.5	$149.2	98.0	74.7	21.9	82.2	$620.5
Kirkos - BayesNet	0.784	95.8	72.4	16.6	63.6	$231.2	97.3	65.3	16.4	79.0	$784.5
Kirkos - ID3	0.783	96.3	85.7	27.3	62.1	$181.8	96.3	73.4	17.9	67.2	$918.9
Kirkos - NeuralNet	0.630	96.0	68.3	15.5	67.2	$236.2	96.8	64.2	15.4	75.6	$861.7
Gaganis - NeuralNet	0.792	97.2	69.9	18.1	76.8	$198.4	97.5	65.5	16.7	80.4	$754.9
Gaganis - LogitReg	0.772	96.7	71.3	17.8	72.1	$208.1	97.2	68.3	17.4	77.0	$788.3
Gaganis - SVM-Lin	0.769	96.7	71.2	17.8	72.1	$208.4	97.3	66.5	16.9	78.7	$775.9

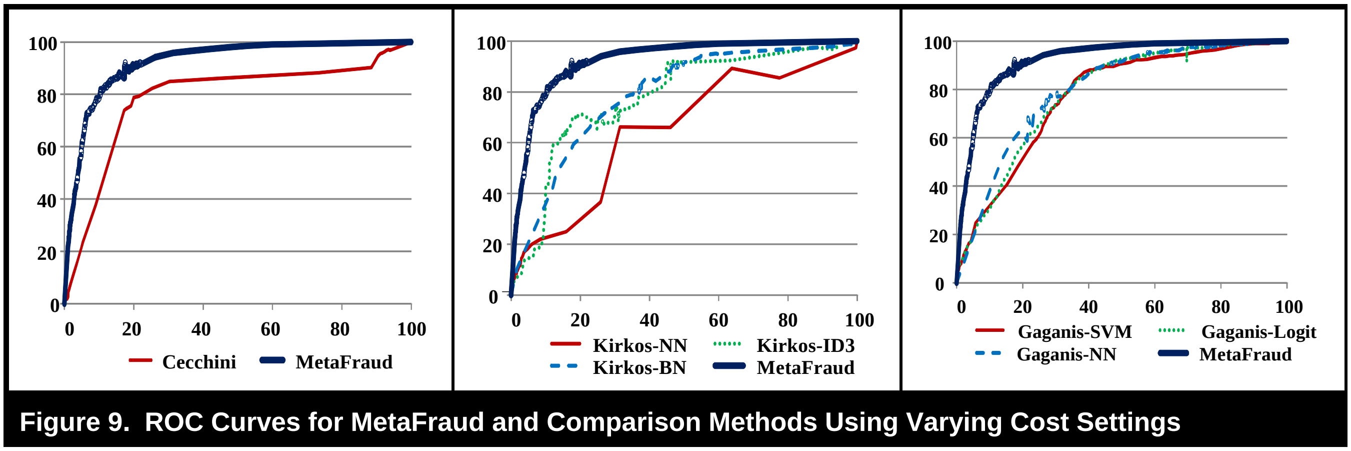


Figure 9. ROC Curves for MetaFraud and Comparison Methods Using Varying Cost Settings

also assessed the error cost per instance (i.e., firm-year) of each method. For the regulator setting, each false positive was assessed a cost of $443,000 (median cost of audit) while false negatives were attributed a cost of $4,100,000 (median cost of financial statement fraud). For the investor setting, the cost of false negatives was computed as a 20 percent drop in the median stock value across firms in our test bed (i.e., 20 percent of $120M, or $24M). The cost of false positives was the opportunity cost of failing to invest in a legitimate firm; this was computed as a 1 percent increase in the median stock value ($1.2M) (Beneish 1999a, 1999b; Cox and Weirich 2002). The error cost numbers depicted are in thousands of dollars. Based on the table, the results for MF were considerably better than those associated with all three comparison methods. MF had higher overall AUC and better legitimate and fraud recalls for both cost settings. These performance gains had a significant financial impact: the errors produced by MF cost approximately $49K to $136K less per firm-year than comparison methods for the regulator setting; MF saved $307K and $225K per firm, respectively, over baselines of auditing all firms and auditing none. From the investor perspective, where low false negative rates are even more critical, MF's error costs were at least $179K per firm less than comparison methods. Across the test bed, MetaFraud's total error costs were $250M better than the best comparison method on the regulator setting and $900M better on the investor setting. Further analysis of the error costs associated with the regulator and investor settings for various components of MetaFraud and comparison methods can be found in Appendices D and E.

Figure 9 shows the ROC curves for MF and the three comparison methods. MF's curve was situated closest to the top left corner, indicating higher ratios of true/false positive rates than comparison methods. The results suggest that MF is capable of providing enhanced detection performance for various cost settings. With respect to the comparison methods, the financial kernel outperformed Kirkos et al. and Gaganis in terms of overall AUC and legit/fraud recall for both cost settings, as evidenced by Table 16. While the best results for the financial kernel were somewhat lower than those attained in the Cecchini et al. study, the overall accuracy results for the Kirkos et al. and Gaganis methods were significantly lower than those reported in their studies (i.e., 15–20 percent). As previously alluded to, this is likely due to the application of these approaches on an entirely different set of data: U.S. firms from various sectors as opposed to Greek manufacturing firms (Barth et al. 2008).

In order to further assess the effectiveness of the MetaFraud framework, we ran MetaFraud using the ratios/measures utilized by Kirkos et al., Gaganis, and Cecchini et al. We called these MF-Kirkos, MF-Gaganis, and MF-Cecchini. For all three feature sets, we derived the industry and organizational context measures from the quarterly and annual statements. For instance, the 7 Gaganis ratios were used to generate 49 annual and 196 quarterly attributes (see Tables 3 and 4 for details). Similarly, the 21 Cecchini et al. measures were used to develop 147 annual and 588 quarterly features. We then ran the combined stack and ASL modules and computed a single performance score for all 991 cost settings (i.e., 1:1 to 1:100), as done in the previous experiment. Table 17 and Figure 10 show the experiment results.

Based on the table, the results for MF-Cecchini, MF-Gaganis, and MF-Kirkos were considerably better than the best comparison results reported in Table 16 for both cost settings. Figure 10 shows the ROC curves for MF-Cecchini, MF-Kirkos, and MF-Gaganis and the comparison techniques. MF is also included to allow easier comparisons with the results from Table 16 and Figure 9. Using MetaFraud improved performance for all three feature sets over the best techniques adopted in prior studies. MF-Cecchini outperformed MF-Kirkos and MF-Gaganis. The lower performance of MF-Kirkos and MF-Gaganis relative to MF-Cecchini was attributable to the fact that the ratios of these methods were less effective on nonmanufacturing firms. Interestingly, the MF-Cecchini ROC curve was very similar to the one generated using MetaFraud with the 12 ratios (i.e., MF). This is because the measures employed by Cecchini et al. (2010) include many of these baseline ratios. Their financial kernel implicitly developed 8 of the 12 ratios (see “Financial Ratios” for details): asset turnover (R2), depreciation index (R5), inventory growth (R7), leverage (R8), operating performance margin (R9), receivables growth (R10), sales growth (R11), and SGE expense (R12). Moreover, they also used variations of asset quality index (R1) and gross margin index (R6). On the flip side, while the larger input feature space for MF-Cecchini did garner slightly better fraud recall rates (relative to MF), the legitimate recall values were 4 to 5 percent lower for both cost settings. Consequently, MF had a better financial impact than MF-Cecchini. Overall, the results demonstrate the efficacy of MetaFraud as a viable mechanism for detecting financial fraud.

H5: MetaFraud Versus Comparison Methods

We conducted paired t-tests to compare the performance of MetaFraud against the seven comparison settings shown in Table 16. We compared cost settings of 5 through 50 in increments of 1 ($n = 46$). MetaFraud significantly outperformed the comparison methods on precision and recall (all p-values < 0.001). We also ran t-tests to compare MF-Kirkos, MF-Gaganis, and MF-Cecchini against their respective settings from Table 16. Once again, using MetaFraud significantly improved performance, with all p-values less than 0.001. The t-test results support H5 and suggest that the proposed meta-learning framework can enhance fraud detection performance over the results achieved by existing methods.

Evaluating MetaFraud in Comparison with Existing Semi-Supervised Learning Methods

In order to demonstrate the effectiveness of the procedural bias improvement mechanisms incorporated by MetaFraud over existing ensemble-based semi-supervised learning methods, we compared MetaFraud against Tri-Training (Zhou and Li 2005). Tri-Training has outperformed existing semi-supervised learning methods on several test beds, across various application domains. It uses an ensemble of three classifiers. In each iteration, the predictions of all test instances where two classifiers *j* and *k* agree are added to the third classifier’s training set (i.e., classifier *i*) with the predicted class labels, provided that the estimated error rates for instances where *j* and *k* agree has improved since the previous iteration. We ran Tri-Training on the base ratios as well as those proposed by the three comparison studies. In order to isolate the impact of MetaFraud’s procedural bias improvement methods, we ran Tri-Training using the context-based features for all four sets of ratios (as done with MetaFraud). For Tri-Training, we evaluated various combinations of classifiers and found that the best results were generally attained when using Bayes Net, Logit, and J48 in conjunction. For these three classifiers, we then ran all 991 cost settings as done in the H5 experiments. Consistent with the settings used by MetaFraud’s ASL algorithm, Tri-Training was run on test instances in chronological order (i.e., all year 2000 instances were evaluated before moving on to the 2001 data).

Table 17. Results for MetaFraud Using Comparison Methods' Ratios

		Regulators (Cost 1:10)					Investors (Cost 1:20)				
		Legit		Fraud		Error	Legit		Fraud		Error
Setting	AUC	Prec.	Rec.	Prec.	Rec.	Cost	Prec.	Rec.	Prec.	Rec.	Cost
MF – Cecchini	0.922	98.2	86.0	33.6	81.7	$116.7	98.3	84.8	32.1	83.4	$485.7
MF – Kirkos	0.851	97.8	80.3	25.9	79.5	$147.1	97.8	71.0	19.5	81.4	$674.9
MF – Gaganis	0.864	97.7	83.0	28.2	77.3	$143.5	98.1	81.1	27.2	81.7	$558.9

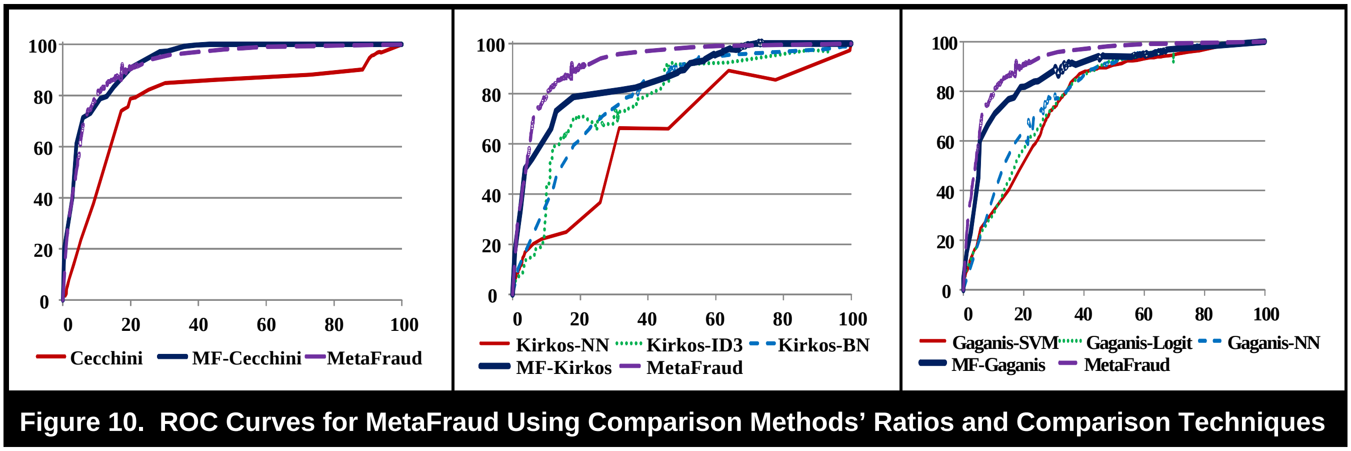


Figure 10. ROC Curves for MetaFraud Using Comparison Methods' Ratios and Comparison Techniques

Table 18. Best Results for Tri-Training Using Four Different Sets of Ratios

		Regulators (Cost 1:10)					Investors (Cost 1:20)				
		Legit		Fraud		Error	Legit		Fraud		Error
Setting	AUC	Prec.	Rec.	Prec.	Rec.	Cost	Prec.	Rec.	Prec.	Rec.	Cost
TT-Cecchini-BN	0.882	97.7	86.0	32.0	76.0	$135.0	97.9	81.2	26.8	79.7	$594.9
TT-Kirkos-Logit	0.769	97.3	70.9	18.6	77.0	$193.4	97.6	68.2	18.0	80.9	$715.5
TT-Gaganis-J48	0.808	97.1	76.3	21.1	73.3	$183.4	97.7	66.8	17.5	81.4	$721.5
TT-Base-BN	0.845	97.5	84.2	29.1	75.1	$145.6	97.6	79.4	24.5	77.5	$657.2

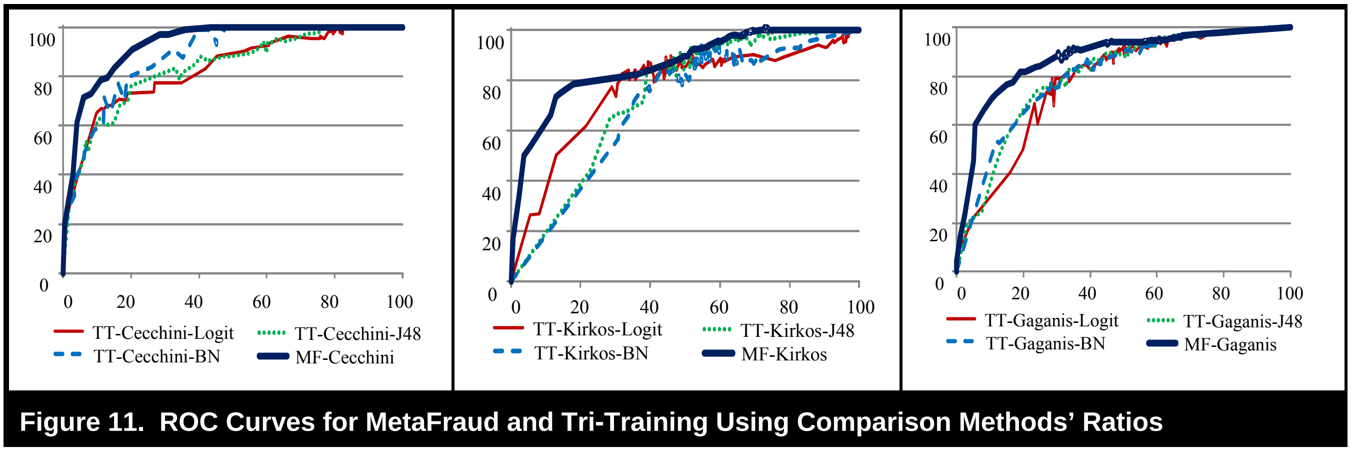


Figure 11. ROC Curves for MetaFraud and Tri-Training Using Comparison Methods' Ratios

Table 18 and Figure 11 show the experiment results, including overall AUC, legit/fraud recall, and error cost per firm-year for both cost settings. Figure 11 shows the ROC curves for MF-Cecchini, MF-Kirkos, and MF-Gaganis in comparison with the three Tri-Training classifiers. Based on the results, it is evident that MetaFraud outperformed Tri-Training, both in terms of best performance results (see the top of Table 16 and Table 17, versus Table 18) and across cost settings. While some settings of Tri-Training improved performance over the original methods run with those features, the performance gains were small in comparison to the improvements garnered by MF-Cecchini, MF-Gaganis, MF-Kirkos, and MF.

H6: MetaFraud Versus Comparison Ensemble-Based Semi-Supervised Learning Methods

Paired t-tests were used to evaluate the performance of MetaFraud relative to the Tri-Training results presented in Table 18. P-values for all four evaluation metrics were significant across the four feature sets (with all p-values < 0.001). The results illustrate how MetaFraud's utilization of an elaborate stacked generalization scheme comprised of several top-level classifiers in conjunction with the ASL algorithm facilitated enhanced procedural bias improvement, and consequently resulted in better fraud detection performance, over existing ensemble-based semi-supervised learning methods.

Discussion

In this study, we proposed a meta-learning framework for enhanced detection of financial fraud. Our research objective for this study was to develop a BI framework that detected fraud from publicly available financial information with demonstratively better performance than that obtained by existing methods. To achieve this objective, we developed a design artifact—the MetaFraud framework—using principles from meta-learning. We also incorporated *ex post* sensitivity analysis as part of the iterative evaluation and refinement process in artifact development. This is consistent with the design science guideline, "design as a search process" (Hevner et al. 2004, p. 83). Our evaluation of the framework, including the detailed results for H1–H6 as well as the analyses presented in the appendices, shows the efficacy of each component of the MetaFraud framework and demonstrates that the complete framework substantially improves performance over existing state-of-the-art methods.

The results from experiment 1 (and H1) demonstrated that incorporating industry-level and organizational context information, whether at the yearly or quarterly level, can improve performance over just using annual-statement-based ratios devoid of context information. In experiment 2, combining yearly and quarterly information yielded the best results as they provide complementary information (H2). Experiment 2 also supported the notion that the ability of stack classifiers to exploit disparate underlying classifiers enables them to improve classification performance (H3). Experiment 3 revealed that the proposed adaptive semi-supervised learning algorithm further improved performance by leveraging the underlying classifications with the highest confidence levels (H4). Experiments 4 and 5 showed that, collectively, the proposed meta-learning framework was able to outperform comparison state-of-the-art methods (H5 and H6).

Each phase of the MetaFraud framework is intended to improve financial fraud detection performance while simultaneously serving as an input refinement mechanism for the ensuing phase of the framework. The first two phases enhanced declarative bias, while the latter two improved procedural bias. For instance, the 12 financial fraud ratios (i.e., the baseline features) were used to generate the yearly and quarterly context-based feature sets (phase A). These features were then used as inputs into the yearly and quarterly context-based classifiers (phase B). The classifications from these base classifiers were used as input features in the combined stack classifiers (phase C). The combined stack classifiers' classifications were used to inform the adaptive semi-supervised learning algorithm (phase D). Collectively, the MetaFraud framework improved overall accuracy by 27 percent on average as compared to the baseline classifiers, and by 7 to 20 percent over state-of-the-art methods, with each phase contributing significantly as evidenced by the hypotheses test results.

Conclusions

Consistent with design science, we used a series of experiments to rigorously test each component of the MetaFraud framework, as well as to compare the framework to existing state-of-the-art methods. The experimental results revealed that the MetaFraud framework was remarkably effective, with legitimate and fraud recall of over 80 percent for different stakeholder cost settings (Table 15). The results also showed that MetaFraud markedly improved performance over existing methods. Overall, the results confirm the viability of using meta-learning methods for enhanced financial fraud detection.

Our research contribution is the development of an innovative framework for financial fraud detection, which integrates BI methods into a meta-learning artifact. The MetaFraud framework encompasses several important aspects that each contribute to its overall effectiveness. The use of organizational and industry contextual information, taken from quarterly and annual data, provides an effective mechanism for expanding the fraud detection feature space. Although stacking and semi-supervised learning have been used previously, we are unaware of any prior work that has similarly integrated an extensive stacked generalization scheme, semi-supervised learning, and adaptive/active learning. The unique combination of these elements in our framework (i.e., through the ASL algorithm) constitutes an important contribution to the fraud detection and BI literature. Specifically, the results of Hypothesis 6 show the effectiveness of this approach over existing methods. MetaFraud outperformed the Tri-Training adaptive-learning approach (Zhou and Li 2005), which has been successfully applied to a dozen problem domains, including credit card approval, credit screening, and web page classification.

Another contribution to the domain of fraud detection is the confidence scores generated by MetaFraud (Figure 4). For instance, the proposed approach was able to attain over 90 percent legitimate and fraud recall on the 70 percent of test instances with the highest confidence scores. These confidence scores can provide a useful decision aid for various stakeholder groups, analogous to the credit-worthiness ratings that are currently available for commercial and government entities. For instance, investors may wish to shy away from firms that are considered fraudulent with a high confidence level. Audit firms can leverage these recommendations in order to better assess the risk associated with new potential clients. Regulators can use such scores to aid in the appropriation and prioritization of investigatory resources.

The MetaFraud framework could also influence the design and development of financial fraud detection systems that integrate predictive and analytical business intelligence technologies, thereby allowing analysts to draw their own conclusions (Bay et al. 2006; Coderre 1999; Virdhagriswaran and Dakin 2006). By combining a rich feature set with a robust classification mechanism that incorporates adaptive learning, meta-learning-based systems could provide fraud risk ratings, real-time alerts, analysis and visualization of fraud patterns (using various financial ratios in the feature sets), and trend analyses of fraud detection patterns over time (utilizing the adaptive learning component). As business intelligence technologies continue to become more pervasive (Watson and Wixom 2007), such systems could represent a giant leap forward, allowing fraud detection tools to perform at their best, when they are combined with human expertise.

References

Abbasi, A., and Chen, H. 2008a. "CyberGate: A Design Framework and System for Text Analysis of Computer-Mediated Communication," *MIS Quarterly* (32:4), pp. 811-837.

Abbasi, A., and Chen, H. 2008b. "Writeprints: A Stylometric Approach to Identity-level Identification and Similarity Detection in Cyberspace," *ACM Transactions on Information Systems* (26:2), no. 7.

Abbasi, A., and Chen, H. 2009. "A Comparison of Fraud Cues and Classification Methods for Fake Escrow Website Detection," *Information Technology and Management* (10), pp. 83-101.

Abbasi, A., Zhang, Z., Zimbra, D., Chen, H., and Nunamaker, Jr., J. F. 2010. "Detecting Fake Websites: The Contribution of Statistical Learning Theory," *MIS Quarterly* (34:3), pp. 435-461.

Accounting Financial and Tax. 2009. "Using Ratios to Detect Fraud and Financial Misstatement," October 10 (http://accounting-financial-tax.com/2009/10/using-ratios-to-detect-fraud-and-financial-misstatement; accessed July 8, 2010).

Albrecht, C. C., Albrecht, W. S., and Dunn, G. 2001. "Can Auditors Detect Fraud: A Review of the Research Evidence," *Journal of Forensic Accounting* (2), pp. 1-12.

Albrecht, W. S., Albrecht, C. C., and Albrecht, C. O. 2004. "Fraud and Corporate Executives: Agency, Stewardship, and Broken Trust," *Journal of Forensic Accounting* (5:1), pp. 109-130.

Albrecht, W. S., Albrecht, C. O., and Albrecht, C. C. 2008. "Current Trends in Fraud and its Detection," *Information Security Journal: A Global Perspective* (17), pp. 2-12.

Ameen, E. C., and Strawser, J. R. 1994. "Investigating the Use of Analytical Procedures: An Update and Extension," *Auditing: A Journal of Theory and Practice* (13:2), pp. 69-76.

Anderson-Lehman, R., Watson, H. J., Wixom, B. H., and Hoffer, J. A. 2004. "Continental Airlines Flies High with Real-Time Business Intelligence," *MIS Quarterly Executive* (3:4), pp. 163-176.

Ando, R. K., and Zhang, T. 2007. "Two-View Feature Generation Model for Semi-Supervised Learning," in *Proceedings of the 24th International Conference on Machine Learning*, Corvallis, OR, June 20-24.

Association of Certified Fraud Examiners. 2010. "2010 Global Fraud Study: Report to the Nations on Occupational Fraud and Abuse," Association of Certified Fraud Examiners, Austin, TX.

Balcan, M. F., Blum, A., and Yang, K. 2005. "Co-Training and Expansion: Towards Bridging Theory and Practice," in *Advances in Neural Information Processing Systems 17*, L. K. Saul, Y. Weiss, and L. Bottou (eds.), Cambridge, MA: MIT Press, pp. 89-96.

Bankruptcydata.com. "20 Largest Public Domain Company Bankrutcy Filings 1980–Present (http://www.bankruptcydata.com/Research/Largest_Overall_All-Time; accessed July 8, 2010).

Barth, M., Landsman, W., and Lang, M. 2008. "International Accounting Standards and Accounting Quality," *Journal of Accounting Research* (46:3), pp. 467-498.

Bay, S., Kumaraswamy, K., Anderle, M. G., Kumar, R., and Steier, D. M. 2006. "Large Scale Detection of Irregularities in Accounting Data," in *Proceedings of the 6th IEEE International Conference on Data Mining*, Hong Kong, December 18-22, pp. 75-86.

Bayes, T. 1958. "Studies in the History of Probability and Statistics: XI. Thomas Bayes' Essay Towards Solving a Problem in the Doctrine of Chances," *Biometrika* (45), pp. 293-295.

Beasley, M. S., Carcello, C. V., Hermanson, D. R., and Lapides, P. D. 2000. "Fraudulent Financial Reporting: Consideration of Industry Traits and Corporate Governance Mechanisms," *Accounting Horizons* (14:4), pp. 441-454.

Beneish, M. D. 1999a. "The Detection of Earnings Manipulation," *Financial Analysts Journal* (55:5), pp. 24-36.

Beneish, M. D. 1999b. "Incentives and Penalties Related to Earnings Overstatements that Violate GAAP," *The Accounting Review* (74:4), pp. 425-457.

Bolton, R. J., and Hand, D. J. 2002. "Statistical Fraud Detection: A Review," *Statistical Science*, (17:3), pp. 235-255.

Brachman, R. J., Khabaza, T., Kloesgen, W., Piatetsky-Shapiro, G., and Simoudis, E. 1996. "Mining Business Databases," *Communications of the ACM* (39:11), pp. 42-48.

Brazdil, P., Giraud-Carrier, C., Soares, C., and Vilalta, R. 2008. *Metalearning: Applications to Data Mining*, Berlin: Springer-Verlag.

Breiman, L. 2001. "Random Forests," *Machine Learning* (45:1), pp. 5-32.

Carson, T. 2003. "Self-Interest and Business Ethics: Some Lessons of the Recent Corporate Scandals," *Journal of Business Ethics* (43:4), pp. 389-394.

Cecchini, M., Aytug, H., Koehler, G., and Pathak, P. 2010. "Detecting Management Fraud in Public Companies," *Management Science* (56:7), pp. 1146-1160.

Chai, W., Hoogs, B., and Verschueren, B. 2006. "Fuzzy Ranking of Financial Statements for Fraud Detection," in *Proceedings of the IEEE International Conference on Fuzzy Systems*, Vancouver, BC, July 16-21, pp. 152-158.

Chan, P. K., Fan, W., Prodromidis, A. L., and Stolfo, S. J. 1999. "Distributed Data Mining in Credit Card Fraud Detection," *IEEE Intelligent Systems* (14:6), pp. 67-74.

Chan, P., and Stolfo, S. 1993. "Toward Parallel and Distributed Learning by Meta-Learning," in *Proceedings of the Knowledge Discovery in Databases Workshop*, pp. 227-240.

Chapelle, O., Schölkopf, B., and Zien, A. 2006. *Semi-Supervised Learning*, Cambridge, MA: MIT Press.

Charles, S. L., Glover, S. M., and Sharp, N. Y. 2010. "The Association Between Financial Reporting Risk and Audit Fees Before and After the Historic Events Surrounding SOX," *Auditing: A Journal of Practice and Theory* (29:1), pp. 15-39.

Chen, W., and Du, Y. 2009. "Using Neural Networks and Data Mining Techniques for the Financial Distress Prediction Model," *Expert Systems with Applications* (36), pp. 4075-4086.

Chung, W., Chen, H., and Nunamaker Jr., J. F. 2005. "A Visual Framework for Knowledge Discovery on the Web: An Empirical Study of Business Intelligence Exploration," *Journal of Management Information Systems* (21:4), pp. 57-84.

Coderre, D. 1999. "Computer-Assisted Techniques for Fraud Detection," *The CPA Journal* (69:8), pp. 57-59.

Cohen, W. W. 1995. "Fast Effective Rule Induction," in *Proceedings of the 12th International Conference on Machine Learning*, Tahoe City, CA, July 9-12, pp. 115-123.

Cox, R. A. K., and Weirich, T. R. 2002. "The Stock Market Reaction to Fraudulent Financial Reporting," *Managerial Auditing Journal* (17:7), pp. 374-382.

Dechow, P., Ge, W., Larson, C., and Sloan, R. 2011. "Predicting Material Accounting Misstatements," *Contemporary Accounting Research* (28:1), pp. 1-16.

Deloitte. 2010. "Deloitte Poll: Majority Expect More Financial Statement Fraud Uncovered in 2010 2011," April 27 (http://www.deloitte.com/view/en_US/us/Services/Financial-Advisory-Services/7ba0852e4de38210VgnVCM200000bb42f00aRCRD.htm; accessed July 8, 2010).

Dikmen, B., and Küçükkocaoğlu, G. 2010. "The Detection of Earnings Manipulation: The Three-Phase Cutting Plane Algorithm Using Mathematical Programming," *Journal of Forecasting* (29:5), pp. 442-466.

Dull, R., and Tegarden, D. 2004. "Using Control Charts to Monitor Financial Reporting of Public Companies," *International Journal of Accounting Information Systems* (5:2), pp. 109-127.

Dybowski, R., Laskey, K. B., Myers, J. W., and Parsons, S. 2003. "Introduction to the Special Issue on the Fusion of Domain Knowledge with Data for Decision Support," *Journal of Machine Learning Research* (4), pp. 293-294.

Dzeroski, S., and Zenko, B. 2004. "Is Combining Classifiers with Stacking Better than Selecting the Best One?" *Machine Learning* (54:3), pp. 255-273.

Fanning, K. M., and Cogger, K. O. 1998. "Neural Network Detection of Management Fraud Using Published Financial Data," *International Journal of Intelligent Systems in Accounting and Finance Management* (7), pp. 21-41.

Fawcett, T., and Provost, F. 1997. "Adaptive Fraud Detection," *Data Mining and Knowledge Discovery* (1), pp. 291-316.

Freund, Y., and Mason, L. 1999. "The Alternating Decision Tree Learning Algorithm," in *Proceedings of the 16th International Conference on Machine Learning*, Bled, Slovenia, June 27-30, pp. 124-133.

Gaganis, C. 2009. "Classification Techniques for the Identification of Falsified Financial Statements: A Comparative Analysis," *International Journal of Intelligent Systems in Accounting and Finance Management* (16), pp. 207-229.

Giraud-Carrier, C., Vilalta, R., and Brazdil, P. 2004. "Introduction to the Special Issue on Meta-Learning," *Machine Learning* (54), pp. 187-193.

Green, B. P., and Calderon, T. G. 1995. "Analytical Procedures and the Auditor's Capacity to Detect Management Fraud," *Accounting Enquiries: A Research Journal* (5:2), pp. 1-48.

Green, B. P., and Choi, J. H. 1997. "Assessing the Risk of Management Fraud through Neural Network Technology," *Auditing* (16:1), pp. 14-28.

Hansen, J. V., and Nelson, R. D. 2002. "Data Mining of Time Series Using Stacked Generalizers," *Neurocomputing* (43), pp. 173-184.

Hevner, A. R., March, S. T., Park, J., and Ram, S. 2004. "Design Science in Information Systems Research," *MIS Quarterly* (28:1), pp. 75-105.

Hu, M. Y., and Tsoukalas, C. 2003. "Explaining Consumer Choice through Neural Networks: The Stacked Generalization Approach," *European Journal of Operational Research* (146:3), pp. 650-660.

Kaminski, K. A., Wetzel, T. S., and Guan, L. 2004. "Can Financial Ratios Detect Fraudulent Financial Reporting," *Managerial Auditing Journal* (19:1), pp. 15-28.

Kinney, W. R., Jr. 1987. "Attention-Directing Analytical Review Using Accounting Ratios: A Case Study," *Auditing: A Journal of Practice and Theory*, Spring, pp. 59-73.

Kirkos, E., Spathis, C., and Manolopoulos, Y. 2007. "Data Mining Techniques for the Detection of Fraudulent Financial Statements," *Expert Systems with Applications* (32), pp. 995-1003.

Kohavi, R. 1996. "Scaling Up the Accuracy of Naïve Bayes Classifiers: A Decision Tree Hybrid," in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, Portland, OR, August 2-4, pp. 202-207.

Kuhn, J. R., Jr., and Sutton, S. G. 2006. "Learning from WorldCom: Implications for Fraud Detection and Continuous Assurance," *Journal of Emerging Technologies in Accounting* (3), pp. 61-80.

Kuncheva, L. I., and Whitaker, C. J. 2003. "Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy," *Machine Learning* (51:2), pp. 181-207.

Langley, P., Zytkow, J. M., Simon, H. A., and Bradshaw, G. L. 1986. "The Search for Regularity: Four Aspects of Scientific Discovery," in *Machine Learning: An Artificial Intelligence Approach*, Vol. II, S. R. Michalski, G. J. Carbonell, and M. T. Mitchell (eds.), San Francisco: Morgan Kaufman, pp. 425-470.

Lin, J. W., Hwang, M. I., and Becker, J. D. 2003. "A Fuzzy Neural Network for Assessing the Risk of Fraudulent Financial Reporting," *Managerial Auditing Journal* (18:8), pp. 657-665.

Lynam, T. R., and Cormack, G. V. 2006. "On-line Spam Filtering Fusion," in *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Seattle, WA, August 6-11, pp. 123-130.

Maletta, M., and Wright, A. 1996. "Audit Evidence Planning: An Examination of Industry Error Characteristics," *Auditing: A Journal of Practice and Theory* (15), pp. 71-86.

March, S. T., and Smith, G. 1995. "Design and Natural Science Research on Information Technology," *Decision Support Systems* (15:4), pp. 251-266.

Markus, M. L., Majchrzak, A., and Gasser, L. 2002. "A Design Theory for Systems That Support Emergent Knowledge Processes," *MIS Quarterly* (26:3), pp. 179-212.

Martin, B. 1995. "Instance-Based Learning: Nearest Neighbor with Generalization," unpublished Master's Thesis, University of Waikato, Computer Science Department, Hamilton, New Zealand.

Matheus, C. J., and Rendell, L. A. 1989. "Constructive Induction on Decision Trees," in *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, San Mateo, CA: Morgan Kaufman, pp. 645-650.

Michalewicz, Z., Schmidt, M., Michalewicz, M., and Chiriac, C. 2007. *Adaptive Business Intelligence*, New York: Springer.

Mitchell, T. M. 1997. *Machine Learning*, New York: McGraw-Hill.

Persons, O. S. 1995. "Using Financial Statement Data to Identify Factors Associated with Fraudulent Financial Reporting," *Journal of Applied Business Research* (11:3), pp. 38-46.

Phua, C., Alahakoon, D., and Lee, V. 2004. "Minority Report in Fraud Detection: Classification of Skewed Data," *ACM SIGKDD Explorations Newsletter* (6:1), pp. 50-59.

Piramuthu, S., Ragavan, H., and Shaw, M. J. 1998. "Using Feature Construction to Improve the Performance of Neural Networks," *Management Science* (44:3), pp. 416-430.

Quinlan, R. 1986. "Induction of Decision Trees," *Machine Learning* (1:1), pp. 81-106.

Rendell, L., Seshu, R., and Tcheng, D. 1987. "Layered Concept-Learning and Dynamically-Variable Bias Management," in *Proceedings of the 10th International Joint Conference on Artificial Intelligence*, San Francisco: Morgan Kaufmann, pp. 308-314.

Shmueli, G., Patel, N., and Bruce, P. 2010. *Data Mining for Business Intelligence* (2nd ed.), Hoboken, NJ: Wiley & Sons.

Sigletos, G., Paliouras, G., Spyropoulos, C. D., and Hatzopoulos, M. 2005. "Combining Information Extraction Systems Using Voting and Stacked Generalization," *Journal of Machine Learning Research* (6), pp. 1751-1782.

Spathis, C. T. 2002. "Detecting False Financial Statements Using Published Data: Some Evidence from Greece," *Managerial Auditing Journal* (17:14), pp. 179-191.

Spathis, C. T., Doumpos, M., and Zopounidis, C. 2002. "Detecting Falsified Financial Statements: A Comparative Study Using Multicriteria Analysis and Multivariate Statistical Techniques," *The European Accounting Review* (11:3), pp. 509-535.

Stempel, J. 2009. "Fraud Seen Growing Faster in Financial Sector," *Reuters*, October 19 (http://www.reuters.com/article/2009/10/19/businesspro-us-fraud-study-idUSTRE59I55920091019).

Storey, V., Burton-Jones, A., Sugumaran, V., and Purao, S. 2008. "CONQUER: A Methodology for Context-Aware Query Processing on the World Wide Web," *Information Systems Research* (19:1), pp. 3-25.

Summers, S. L., and Sweeney, J. T. 1998. "Fraudulently Misstated Financial Statements and Insider Trading: An Empirical Analysis," *The Accounting Review* (73:1), pp. 131-146.

Tian, Y., Weiss, G. M., and Ma, Q. 2007. "A Semi-Supervised Approach for Web Spam Detection Using Combinatorial Feature-Fusion," in *Proceedings of the ECML Graph Labeling Workshops' Web Spam Challenge*, Warsaw, Poland, September 17-21, pp. 16-23.

Ting, K. M., and Witten, I. H. 1997. "Stacked Generalization: When Does It Work?" in *Proceedings of the 15th Joint International Conference on Artificial Intelligence*, San Francisco: Morgan Kaufmann., pp. 866-871.

Tsoumakas, G., Angelis, L., and Vlahavas, I. 2005. "Selective Fusion of Heterogeneous Classifiers," *Intelligent Data Analysis* (9), pp. 511-525.

Vapnik, V. 1999. *The Nature of Statistical Learning Theory*, Berlin: Springer-Verlag.

Vercellis, C. 2009. *Business Intelligence: Data Mining and Optimization for Decision Making*, Hoboken, NJ: Wiley.

Vilalta, R., and Drissi, Y. 2002. "A Perspective View and Survey of Meta-Learning," *Artificial Intelligence Review* (18), pp. 77-95.

Vilalta, R., Giraud-Carrier, C., Brazdil, P., and Soares, C. 2004. "Using Meta-Learning to Support Data Mining," *International Journal of Computer Science and Applications* (1:1), pp. 31-45.

Virdhagriswaran, S., and Dakin, G. 2006. "Camouflaged Fraud Detection in Domains with Complex Relationships," in *Proceedings of the 12th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, New York: ACM, pp. 941-947.

Walls, J. G., Widmeyer, G. R., and El Sawy, O. A. 1992. "Building an Information System Design Theory for Vigilant EIS," *Information Systems Research* (3:1), pp. 36-59.

Watson, H. J., and Wixom, B. H. 2007. "The Current State of Business Intelligence," *IEEE Computer* (40:9), pp. 96-99.

Witten, I. H., and Frank, E. 2005. *Data Mining: Practical Machine Learning Tools and Techniques* (2nd ed.), San Francisco: Morgan Kaufmann.

Wolpert, D. H. 1992. "Stacked Generalization," *Neural Networks* (6), pp. 241-259.

Yue, D., Wu, X., Wang, Y., and Li, Y. 2007. "A Review of Data Mining-Based Financial Fraud Detection Research," in *Proceedings of the International Conference on Wireless Communications Networking and Mobile Computing*, Shanghai, September 21-25, pp. 5519-5522.

Zekany, K., Braun, L., and Warder, Z. 2004. "Behind Closed Doors at WorldCom: 2001," *Issues in Accounting Education* (19:10), pp. 101-117.

Zhao, H., Sinha, A. P., and Ge, W. 2009. "Effects of Feature Construction on Classification Performance: An Empirical Study in Bank Failure Prediction," *Expert Systems with Applications* (36), pp. 2633-2644.

Zhou, Y., and Goldman, S. 2004. "Democratic Co-learning," in *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence*, pp. 594-202.

Zhou, Z. and Li, M. 2005. "Tri-Training: Exploiting Unlabeled Data Using Three Classifiers," *IEEE Transactions on Knowledge and Data Engineering* (17:11), pp. 1529-1541.

About the Authors

Ahmed Abbasi is an assistant professor of Information Technology in the McIntire School of Commerce at the University of Virginia. He received his Ph.D. in MIS from the University of Arizona and an M.B.A. and B.S. in Information Technology from Virginia Tech. His research interests include fraud detection, online security, and text mining. Ahmed's projects on Internet fraud and social media analytics have been funded by the National Science Foundation. His research has appeared, among other outlets, in *MIS Quarterly*, *Journal of Management Information Systems*, *IEEE Transactions on Knowledge and Data Engineering*, *IEEE Intelligent Systems*, and *ACM Transactions on Information Systems*. He is a member of the AIS and IEEE.

Conan C. Albrecht is an associate professor in the Information Systems Department at the Marriott School of Management, Brigham Young University. He received his Ph.D. in MIS from the University of Arizona and his Master's of Accountancy from Brigham Young University. His research is focused on algorithms and data structures that support computer-based fraud detection. Conan has been involved in fraud detection systems at British Petroleum, the World Bank, and the State of North Carolina, and he has trained auditors and fraud detectives for the Association of Certified Fraud Examiners. His work is published in *Computational Intelligence*, *Decision Support Systems*, *Communications of the ACM*, *Information & Management*, and *IEEE Transactions on Systems, Man, and Cybernetics*. Conan is the author of the Picalo open source fraud detection system, available at http://www.picalo.org.

Anthony Vance is as an assistant professor in the Information Systems Department at the Marriott School of Management, Brigham Young University. He has earned Ph.D. degrees in Information Systems from Georgia State University, USA; the University of Paris–Dauphine, France; and the University of Oulu, Finland. He received a Master's of Information Systems Management from Brigham Young University, during which he was also enrolled in the IS Ph.D. preparation program. His previous experience includes working as a visiting research professor in the Information Systems Security Research Center at the University of Oulu, where he remains a research fellow. He also worked as an information security consultant and fraud analyst for Deloitte. His work is published in *MIS Quarterly*, *Journal of Management Information Systems*, *European Journal of Information Systems*, *Journal of the American Society for Information Science and Technology*, *Information & Management*, *Journal of Organizational and End User Computing*, and *Communications of the AIS*. His research interests are information security and trust in information systems.

James V. Hansen is the J. Owen Cherrington Professor of Information Systems in the Information Systems Department at the Marriott School of Management, Brigham Young University. He received his Ph.D. in Computer Science (Machine Learning) from the University of Washington, Seattle. He is an associate editor for *IEEE Intelligent Systems* and a former associate editor for *Accounting Review*. His research focus is on machine learning and data mining, with related publications appearing in *Management Science*, *MIS Quarterly*, *ACM Computing Surveys*, *Computational Intelligence*, *IEEE Transactions on Neural Networks and Learning*, *IEEE Transactions on Knowledge and Data Engineering*, *Communications of the ACM*, *Decision Support Systems*, and *Decision Sciences*, among others.